A LINESEARCH-BASED DERIVATIVE-FREE APPROACH FOR NONSMOOTH CONSTRAINED OPTIMIZATION*

G. FASANO†, G. LIUZZI‡, S. LUCIDI§, AND F. RINALDI¶

Abstract. In this paper, we propose new linesearch-based methods for nonsmooth constrained optimization problems when first-order information on the problem functions is not available. In the first part, we describe a general framework for bound-constrained problems and analyze its convergence toward stationary points, using the Clarke–Jahn directional derivative. In the second part, we consider inequality constrained optimization problems where both objective function and constraints can possibly be nonsmooth. In this case, we first split the constraints into two subsets: difficult general nonlinear constraints and simple bound constraints on the variables. Then, we use an exact penalty function to tackle the difficult constraints and we prove that the original problem can be reformulated as the bound-constrained minimization of the proposed exact penalty function. Finally, we use the framework developed for the bound-constrained case to solve the penalized problem. Moreover, we prove that every accumulation point, under standard assumptions on the search directions, of the generated sequence of iterates is a stationary point of the original constrained problem. In the last part of the paper, we report extended numerical results on both bound-constrained and nonlinearly constrained problems, showing that our approach is promising when compared to some state-of-the-art codes from the literature.

Key words. derivative-free optimization, Lipschitz optimization, exact penalty functions, inequality constrained optimization, stationarity conditions

AMS subject classifications. 90C30, 90C56, 65K05, 49J52

DOI. 10.1137/130940037

1. Introduction. In this paper, we consider the optimization of a nonsmooth function $f : \mathbb{R}^n \to \mathbb{R}$ over a feasible set defined by lower and upper bounds on the variables and, possibly, by nonlinear and nonsmooth inequality constraints $g : \mathbb{R}^n \to \mathbb{R}^m$, namely,

$$
\begin{aligned}
\min \quad & f(x) \\
\text{s.t.} \quad & g(x) \leq 0, \\
& l \leq x \leq u,
\end{aligned}
$$

where $l, u \in \mathbb{R}^n$, and $l < u$. We observe that, to our purposes, the fact that l and u must be finite might be relaxed by suitable assumptions on the objective function of the problem. We assume that the problem functions (though nonsmooth) are Lipschitz continuous and that first-order information is unavailable or impractical to obtain (e.g., when problem functions are expensive to evaluate or somewhat noisy).

†Dipartimento di Management, Università Ca'Foscari Venezia, San Giobbe, Cannaregio 873, 30121 Venice, Italy (fasano@unive.it).

‡Istituto di Analisi dei Sistemi ed Informatica "A. Ruberti," Consiglio Nazionale delle Ricerche, Viale Manzoni 30, 00185 Rome, Italy (giampaolo.liuzzi@iasi.cnr.it).

§Dipartimento di Ingegneria Informatica Automatica e Gestionale "A. Ruberti," "Sapienza" Università di Roma, Via Ariosto 25, 00185 Rome, Italy (lucidi@dis.uniroma1.it).

¶Dipartimento di Matematica, Università di Padova, Via Trieste 63, 35121 Padua, Italy (rinaldi@math.unipd.it).

Such optimization problems encompass many real-world problems arising in different fields, like computational mathematics, physics, and engineering, and present a twofold difficulty. On the one hand, problem functions are typically of the black-box type, so that first-order information is unavailable; on the other hand, the functions present a certain level of nonsmoothness (see, e.g., [3], [11], and [24]).

In [4] and [5], the mesh adaptive direct search (MADS) class of algorithms is introduced, where a dense set of directions is generated and combined with an extreme barrier approach, in order to provide a general and flexible framework for nonsmooth constrained problems. In [2], the use of a deterministic scheme for the generation of a dense set of search directions is proposed, thus defining the ORTHOMADS method. A different way to handle the constraints within MADS-type algorithms is proposed in [6], where the authors combine a filter-based strategy [17] with a progressive barrier approach. In [46], the most general result for direct search methods of this type is given (for functions directionally Lipschitz) and, moreover, it is shown that integer lattices can be replaced by sufficient decrease when using polling directions dense in the unit sphere.

In [14], it is proved that the efficiency of direct search methods (e.g., MADS), when applied to nonsmooth problems, can be improved by using simplex gradients to order poll directions.

In this work, we extend the linesearch approach with sufficient decrease for unconstrained minimization problems in [18, 35] to the case of nonsmooth bound-constrained and nonlinearly constrained minimization problems. This approach gives us a twofold achievement. On the one hand, by means of the sufficient decrease we can avoid the use of integer lattices. On the other hand, the extrapolation phase allows us to better exploit a descent direction and hence to characterize all the limit points of the sequence of iterates, under some density assumptions on the search directions.

In the first part of this paper, we describe a general framework for solving bound-constrained nonsmooth optimization problems. The approach, called DFN_{simple} (derivative-free nonsmooth simple), uses a projected linesearch procedure. For this simple algorithm we can prove convergence to stationary points of the problem in the Clarke–Jahn sense [22] (see also Definition 2.3 below). Then, we propose an improved version of the algorithm, namely, CS-DFN (coordinate search derivative-free nonsmooth), which further performs linesearches along the coordinate directions.

In the second part, we focus on nonlinearly constrained problems. We assume that two different classes of constraints exist, namely, difficult general nonsmooth constraints ($g(x) \leq 0$) and simple bound constraints on the problem variables ($l \leq x \leq u$). The main idea consists of getting rid of the nonlinear constraints by means of an exact penalty approach. Therefore, we construct a merit function that penalizes the general nonlinear inequality constraints and we resort to the minimization of the penalty function subject to the simple bound constraints. We acknowledge that the idea of only penalizing the nonlinear constraints is not new in the context of derivative-free optimization. Indeed, for smooth problems, the same handling of easy constraints has been previously adopted in several papers like, for instance, [28, 33] (for a combination of inequality and bound constraints) and [25] (for a combination of general and linear constraints). For nonsmooth problems, in [32] finite minimax problems are considered with explicit handling of linear inequality constraints; in [20] the idea of projecting onto simple constraints is proposed for Lipschitz problems. Following this approach, we can resort to the framework developed for the bound-constrained case

and define an algorithm (which is called DFN_{con}) to tackle nonlinearly constrained nonsmooth problems. We are able to prove that the new bound-constrained problem is to a large extent equivalent to the original problem and that the sequence generated by means of the described approach converges to stationary points of the original problem, in the sense that every accumulation point is stationary for the constrained problem.

In the last part of the paper, an extensive numerical experience (on 142 bound-constrained and 296 nonlinearly constrained problems) is carried out. We first test two versions of the DFN_{simple} algorithm, obtained by embedding into the scheme two different pseudorandom sequences to generate the sequence of search directions. In particular, we compare the Halton [19] and Sobol sequences [8, 43] within our method. Then we analyze the performances of both our methods DFN_{simple} and CS-DFN, and we compare CS-DFN with two state-of-the-art solvers on the test set of bound-constrained nonsmooth problems. Finally, we focus on nonlinearly constrained problems. In this case, we compare our code DFN_{con} with two well-known codes on the test set of nonsmooth constrained problems. The codes DFN_{simple}, CS-DFN, and DFN_{con} are freely available for download at http://www.dis.uniroma1.it/~lucidi/DFL.

The paper has the following structure. In section 2, we analyze the approach for the bound-constrained case. In section 3, we extend the approach to nonlinearly constrained problems. The numerical results are reported in section 4. We summarize our conclusions in section 5, and an appendix completes the paper, including auxiliary results.

1.1. Notation and definitions. Given a vector $v \in \mathbb{R}^n$, a subscript will be used to denote either the ith of its entries v_i or the fact that it is an element of an infinite sequence of vectors $\{v_k\}$. In the case of possible misunderstanding or ambiguities, the ith component of a vector will be denoted by $(v)_i$. By $\|\cdot\|$ we indicate the Euclidean norm. We denote by v^j the generic jth element of a finite set of vectors, and in particular $e^1, \ldots, e^n$ represent the coordinate unit vectors. Given two vectors $a, b \in \mathbb{R}^n$, we indicate with $y = \max\{a,b\}$ ($y = \min\{a,b\}$) the vector such that $y_i = \max\{a_i, b_i\}$ ($y_i = \min\{a_i, b_i\}$), $i = 1, \ldots, n$. Furthermore, given a vector $v \in \mathbb{R}^n$ we denote by $v^+ = \max\{0, v\}$. By $S(0,1)$ we indicate the unit sphere with center in the origin, i.e., $S(0,1) = \{x \in \mathbb{R}^n : \|x\| = 1\}$. Further, $[x]_{[l,u]} = \max\{l, \min\{u, x\}\}$ denotes the orthogonal projection over the set $\{x \in \mathbb{R}^n : l \leq x \leq u\}$, and $\overset{\circ}{\mathcal{C}}$ is the interior of set $\mathcal{C}$. From [41, Theorem 3.3], we give the following definition.

DEFINITION 1.1 (convex hull). *Given a collection of finitely many nonempty convex sets* $\Gamma = \{A_i \subseteq \mathbb{R}^n : i = 1, \ldots, m\}$, $Co(\Gamma)$ *indicates its convex hull, namely,*

$$Co(\Gamma) = \bigcup \left\{ \sum_{i=1}^{m} \lambda_i A_i \right\},$$

where the union is taken considering all coefficients $\lambda_i \geq 0$, $i = 1, \ldots, m$, *with* $\sum_{i=1}^{n} \lambda_i = 1$.

From [9], we recall the definition of Clarke stationarity.

DEFINITION 1.2 (Clarke stationarity). *Given the unconstrained problem* $\min_{x \in \mathbb{R}^n} f(x)$, *a point* $\bar{x}$ *is a Clarke stationary point if* $0 \in \partial f(\bar{x})$, *where* $\partial f(x) = \{s \in \mathbb{R}^n : f^{Cl}(x; d) \geq d^T s, \forall d \in \mathbb{R}^n\}$ *is the generalized gradient of* f *at* x, *and*

$$(1.1) \qquad f^{Cl}(x; d) = \limsup_{y \to x, \, t \downarrow 0} \frac{f(y + td) - f(y)}{t}.$$

Recalling the convergence analysis carried out in [2, 5] for derivative-free nonsmooth optimization, the directions used in the optimization algorithm have to satisfy a condition that is intimately connected with the nonsmoothness of the problem itself. Indeed, since the cone of feasible descent directions can be arbitrarily narrow, there is no theoretical guarantee that by using finite sets of search directions one of them yields a negative Clarke directional derivative (1.1) of the objective function (see, e.g., [11, 24]).

Thus, we finally recall the well-known definition of a dense subsequence of directions that will be used throughout the paper.

DEFINITION 1.3 (dense sequence). *Let K be an infinite subset of indices (possibly $K = \{0, 1, \ldots\}$). The subsequence of normalized directions $\{d_k\}_K$ is said to be dense in the unit sphere $S(0, 1)$ if for any $\bar{d} \in S(0, 1)$ and for any $\epsilon > 0$ there exists an index $k \in K$ such that $\|d_k - \bar{d}\| \le \epsilon$.*

2. The bound-constrained case. In this section we consider the bound-constrained problem

$$(2.1) \qquad \begin{aligned} &\min \quad f(x) \\ &\text{s.t.} \quad x \in X, \end{aligned}$$

where we indicate by X the set of bound constraints on the variables, i.e.,

$$X = \{x \in \mathbb{R}^n : l \le x \le u\},$$

and f is Lipschitz continuous. We recall that, since l and u are both finite, set X is compact. For points in the feasible set X we address also the definition of cone of feasible directions, as follows.

DEFINITION 2.1 (cone of feasible directions). *Given problem (2.1) and any point $x \in X$,*

$$D(x)$$
$$= \{d \in \mathbb{R}^n : d_i \ge 0 \text{ if } x_i = l_i, \ d_i \le 0 \text{ if } x_i = u_i, d_i \in \mathbb{R} \text{ if } l_i < x_i < u_i, \ i = 1, \ldots, n\}$$

is the cone of feasible directions at x with respect to X.

We also report a technical proposition whose proof can be found in [29, Proposition 2.3].

PROPOSITION 2.2. *Given problem (2.1), let $\{x_k\} \subset X$ for all k and $\{x_k\} \to \bar{x}$ for $k \to \infty$. Then, for k sufficiently large,*

$$D(\bar{x}) \subseteq D(x_k).$$

The necessary optimality conditions for problem (2.1) can be characterized in terms of the Clarke–Jahn generalized directional derivative of the objective function, instead of

using definition (1.1). Given a point $x \in X$, the Clarke–Jahn generalized directional derivative of function f along direction $d \in D(x)$ is given by (see [22, section 3.5])

$$(2.2) \qquad f^{\circ}(x; d) = \limsup_{\substack{y \to x,\, y \in X \\ t \downarrow 0,\, y + td \in X}} \frac{f(y + td) - f(y)}{t}.$$

From [22, Theorem 4.14] we recall that every local minimum of problem (2.1) satisfies the following definition.

DEFINITION 2.3 (Clarke–Jahn stationarity). *Given problem* (2.1), $x^{\star}$ *is a Clarke–Jahn stationary point if*

$$(2.3) \qquad f^{\circ}(x^{\star}; d) \geq 0 \qquad \forall\, d \in D(x^{\star}).$$

We propose in the next sections two algorithms having different performances on the nonsmooth bound-constrained problem (2.1).

2.1. A simple derivative-free algorithm. As discussed in the introduction, even in the simpler case of bound constraints, since the objective function f is possibly not continuously differentiable on X, a finite number of search directions is not sufficient to investigate the local behavior of $f(x)$ on X [24, section 6.4]. Hence, recalling [5], it is necessary to assume density properties on particular subsequences of the search directions. In this way we are able to prove convergence to stationary points of problem (2.1).

To this purpose, here we propose a very simple derivative-free algorithm for solving the nonsmooth problem (2.1), namely, DFN$_{simple}$.

In this algorithm we use a predefined sequence of search directions $\{d_k\}$. Then, we investigate the behavior of the function $f(x)$ along the direction d_k by means of the linesearch procedure *projected continuous search*. Given the current iterate x_k at step k, the latter procedure first evaluates the function at $[x_k \pm \tilde{\alpha}_k d_k]_{[l,u]}$. In case a sufficient reduction of the function value is obtained, an extrapolation along the search direction is performed, so that a suitable steplength α_k is computed, and is used as a tentative steplength for the next iteration, i.e., $\tilde{\alpha}_{k+1} = \alpha_k$. On the other hand, if at $[x_k \pm \tilde{\alpha}_k d_k]_{[l,u]}$ we do not obtain a sufficient reduction of the function value, then the tentative steplength at the next iteration is suitably reduced by a scale factor, i.e., $\tilde{\alpha}_{k+1} = \theta \tilde{\alpha}_k$, $\theta \in (0,1)$. More formally, the resulting algorithm and the proposed linesearch procedure are summarized in the next schemes.

ALGORITHM DFN$_{simple}$.

Input. $\theta \in (0,1)$, $x_0 \in X$, $\tilde{\alpha}_0 > 0$, a sequence $\{d_k\}$ such that $d_k \in \mathbb{R}^n$, $\|d_k\| = 1$, for all k.

For $k = 0, 1, \ldots$
  Compute α_k and $\tilde{d}_k$ by the projected continuous search $(\tilde{\alpha}_k, x_k, d_k; \alpha_k, \tilde{d}_k)$.
  If $(\alpha_k = 0)$ **then** $\tilde{\alpha}_{k+1} = \theta \tilde{\alpha}_k$ and $\tilde{x}_k = x_k$
  else $\tilde{\alpha}_{k+1} = \alpha_k$ and $\tilde{x}_k = [x_k + \alpha_k \tilde{d}_k]_{[l,u]}$.
  Find $x_{k+1} \in X$ such that $f(x_{k+1}) \leq f(\tilde{x}_k)$.
End For

Output. The sequences $\{x_k\}$, $\{\alpha_k\}$, and $\{\tilde{\alpha}_k\}$.

PROJECTED CONTINUOUS SEARCH $(\tilde{\alpha}, y, p; \alpha, p^+)$.

Data. $\gamma > 0$, $\delta \in (0,1)$.

Step 0. Set $\alpha = \tilde{\alpha}$.

Step 1. If $f([y + \alpha p]_{[l,u]}) \leq f(y) - \gamma \alpha^2$ **then** set $p^+ = \quad p$ and go to Step 4.

Step 2. If $f([y - \alpha p]_{[l,u]}) \leq f(y) - \gamma \alpha^2$ **then** set $p^+ = -p$ and go to Step 4.

Step 3. Set $\alpha = 0$, **return** α and $p^+ = p$.

Step 4. Let $\beta = \alpha/\delta$.

Step 5. If $f([y + \beta p^+]_{[l,u]}) > f(y) - \gamma \beta^2$ **return** α, p^+.

Step 6. Set $\alpha = \beta$ and go to Step 4.

For clarity, we note that the projected continuous search procedure takes in input $\tilde{\alpha}, y$, and p (that is, the arguments before the semicolon) and gives in output α and p^+ (that is, the arguments after the semicolon).

It is worth noting that in Algorithm DFN$_{simple}$ the next iterate x_{k+1} is required to satisfy $f(x_{k+1}) \leq f(\tilde{x}_k)$. This allows us in principle to compute x_{k+1} by minimizing suitable approximating models of the objective function, thus possibly improving the efficiency of the overall scheme.

Furthermore, since we are interested in studying the asymptotic convergence properties of DFN$_{simple}$, its formal definition does not include a stopping condition. We note that this is in accordance with most of the papers concerning convergence of derivative-free methods; see, e.g., [6, 10, 12, 13, 16, 44], among others. We refer the reader to section 4 for a practical stopping condition.

In the following results we analyze the global convergence properties of Algorithm DFN$_{simple}$. In particular, in the next proposition we prove that the projected continuous search cannot cycle.

PROPOSITION 2.4. *The projected continuous search cannot infinitely cycle between Step* 4 *and Step* 6.

Proof. Let us consider the projected continuous search. We proceed by contradiction assuming that an infinite monotonically increasing sequence of positive numbers $\{\beta_j\}$ exists such that

$$f([y + \beta_j p^+]_{[l,u]}) \leq f(y) - \gamma \beta_j^2.$$

The above relation contradicts the fact that X is compact, by definition, and that function f is continuous, thus concluding the proof. $\square$

Now, in the following proposition we prove that the stepsizes computed by the projected continuous search procedure eventually go to zero.

PROPOSITION 2.5. *Let $\{\alpha_k\}$, $\{\tilde{\alpha}_k\}$ be the sequences produced by Algorithm DFN$_{simple}$; then*

$$(2.4) \qquad \lim_{k \to \infty} \max\{\alpha_k, \tilde{\alpha}_k\} = 0.$$

Proof. We split the iteration sequence $\{k\}$ into two sets K_1, K_2, with $K_1 \cup K_2 = \{k\}$ and $K_1 \cap K_2 = \emptyset$. We denote
- K_1 the set of iterations when $\tilde{\alpha}_{k+1} = \alpha_k$,
- K_2 the set of iterations when $\tilde{\alpha}_{k+1} = \theta \tilde{\alpha}_k$ and $\alpha_k = 0$.

Note that K_1 and K_2 cannot both be finite. Then we analyze the following two cases, K_1 infinite (Case I) and K_1 finite (Case II).

Case I. Since K_1 is infinite, then the instructions of the algorithm imply, for $k \in K_1$,

$$(2.5) \qquad f(x_{k+1}) \leq f([x_k + \alpha_k \tilde{d}_k]_{[l,u]}) \leq f(x_k) - \gamma \alpha_k^2.$$

Taking into account the compactness of X and the continuity of f, we get from the above relation that $\{f(x_k)\}$ tends to a limit $\bar{f}$. Then, by (2.5), it follows that

$$(2.6) \qquad \lim_{k \to \infty, k \in K_1} \alpha_k = 0,$$

which also implies

$$(2.7) \qquad \lim_{k \to \infty, k \in K_1} \tilde{\alpha}_k = 0.$$

Case II. Recall that K_1 is finite in this case, so that set K_2 must be infinite. Since, by definition, $\alpha_k = 0$, $k \in K_2$, we have

$$(2.8) \qquad \lim_{k \to \infty, k \in K_2} \alpha_k = 0.$$

Then, let $m_k < k$ be the largest integer such that $m_k \in K_1$. By the instructions of the algorithm, we can write

$$(2.9) \qquad \tilde{\alpha}_{k+1} = \theta^{k-m_k} \tilde{\alpha}_{m_k}.$$

Note that in case the index m_k does not exist (when K_1 is empty), we set $m_k = 0$. When $k \to \infty$ and $k \in K_2$, we have only the following two cases: either $m_k \to \infty$ (i.e., K_1 is an infinite subset) or $(k - m_k) \to \infty$ (i.e., K_1 is finite). Therefore, (2.7) and (2.9) along with $\theta \in (0, 1)$ give

$$(2.10) \qquad \lim_{k \to \infty, k \in K_2} \tilde{\alpha}_k = 0.$$

Relations (2.6), (2.7), (2.8), and (2.10) yield (2.4), thus concluding the proof. $\square$

Using the latter result we can provide the next technical lemma, which will be necessary to prove the main global convergence result for algorithm DFN_{simple}. This lemma shows that the projection operator does not sensibly deteriorate the asymptotic properties of the directions d_k. More precisely, performing a steplength η_k along d_k and assuming that η_k goes to zero, it results that eventually the new point $[x_k + \eta_k d_k]_{[l,u]}$ differs from x_k and the scaled actual step $([x_k + \eta_k d_k]_{[l,u]} - x_k)/\eta_k$ enjoys the same asymptotic properties of d_k.

LEMMA 2.6. *Let $\{x_k\}$ be the sequence produced by Algorithm DFN_{simple}, let $\{d_k\}$ be the sequence of search directions used by DFN_{simple}, and let $\{\eta_k\}$ be a sequence such that $\eta_k > 0$ for all k. Further, let K be a subset of indices such that*

$$(2.11) \qquad \lim_{k \to \infty, k \in K} x_k = \bar{x},$$

$$(2.12) \qquad \lim_{k \to \infty, k \in K} d_k = \bar{d},$$

$$(2.13) \qquad \lim_{k \to \infty, k \in K} \eta_k = 0$$

with $\bar{x} \in X$ and $\bar{d} \in D(\bar{x})$, $\bar{d} \neq 0$. Then,

(i) *for all $k \in K$ sufficiently large,*

$$[x_k + \eta_k d_k]_{[l,u]} \neq x_k;$$

(ii) *the following limit holds:*

$$\lim_{k \to \infty, k \in K} v_k = \bar{d},$$

where

(2.14) $$v_k = \frac{[x_k + \eta_k d_k]_{[l,u]} - x_k}{\eta_k}.$$

Proof. In order to prove items (i) and (ii), let us recall that

$$[x_k + \eta_k d_k]_{[l,u]} = \max\{l, \min\{u, (x_k + \eta_k d_k)\}\}.$$

Now we show that for $k \in K$ sufficiently large

(2.15) $$[x_k + \eta_k d_k]_{[l,u]} \neq x_k.$$

By contradiction, let us assume that for $k \in K$ sufficiently large, we have

(2.16) $$[x_k + \eta_k d_k]_{[l,u]} = x_k.$$

Since by assumption $\bar{d} \neq 0$, an index i with $\bar{d}_i \neq 0$ exists and one of the following three cases holds:

(1) $\bar{x}_i = l_i$ (which implies $\bar{d}_i > 0$). We can write

$$([x_k + \eta_k d_k]_{[l,u]})_i = \max\{l_i, (x_k + \eta_k d_k)_i\};$$

since x_k is feasible and (2.12) holds, for k sufficiently large we have

$$\max\{l_i, (x_k + \eta_k d_k)_i\} > \max\left\{l_i, \left(x_k + \frac{\eta_k}{2}\bar{d}\right)_i\right\},$$

so that, by (2.13) and $\eta_k > 0$, we get

(2.17) $$\max\left\{l_i, \left(x_k + \frac{\eta_k}{2}\bar{d}\right)_i\right\} = \left(x_k + \frac{\eta_k}{2}\bar{d}\right)_i \neq (x_k)_i.$$

(2) $\bar{x}_i = u_i$ (which implies $\bar{d}_i < 0$). We can write

$$([x_k + \eta_k d_k]_{[l,u]})_i = \min\{u_i, (x_k + \eta_k d_k)_i\};$$

since x_k is feasible and (2.12) holds, for k sufficiently large we have

$$\min\{u_i, (x_k + \eta_k d_k)_i\} < \min\left\{u_i, \left(x_k + \frac{\eta_k}{2}\bar{d}\right)_i\right\},$$

so that by (2.13) and $\eta_k > 0$ we get

(2.18) $$\min\left\{u_i, \left(x_k + \frac{\eta_k}{2}\bar{d}\right)_i\right\} = \left(x_k + \frac{\eta_k}{2}\bar{d}\right)_i \neq (x_k)_i.$$

(3) $l_i < \bar{x}_i < u_i$ (which implies $\bar{d}_i \neq 0$). We can write

$$([x_k + \eta_k d_k]_{[l,u]})_i = (x_k + \eta_k d_k)_i;$$

since x_k is feasible and (2.12) holds, for k sufficiently large we have

(2.19) $$(x_k + \eta_k d_k)_i \neq (x_k)_i.$$

Then, by (2.17), (2.18), and (2.19) we have a contradiction with (2.16), which proves (i).

Now, we recall definition (2.14) and note that, by (2.15), the vector v_k is eventually nonzero. By the definition of the vector v_k, we have for its ith entry

(2.20) $$(v_k)_i = \frac{\max\{l_i, \min\{u_i, (x_k + \eta_k d_k)_i\}\} - (x_k)_i}{\eta_k}$$

(2.21) $$= \frac{\min\{u_i, \max\{l_i, (x_k + \eta_k d_k)_i\}\} - (x_k)_i}{\eta_k}.$$

Now, let us distinguish among the following three cases for k sufficiently large and $k \in K$:

(1) $\bar{x}_i = l_i$. Then by (2.20) we have

$$(v_k)_i = \frac{\max\{l_i, (x_k + \eta_k d_k)_i\} - (x_k)_i}{\eta_k}$$

and recalling that whenever $\bar{x}_i = l_i$ it must be $\bar{d}_i \geq 0$, we distinguish two subcases:

a) when $\bar{d}_i > 0$, then $(v_k)_i = \max\left\{\frac{l_i - (x_k)_i}{\eta_k}, (d_k)_i\right\} = (d_k)_i$;

b) when $\bar{d}_i = 0$, then

$$\lim_{k\to\infty, k\in K} (v_k)_i = \lim_{k\to\infty, k\in K} \max\left\{\frac{l_i - (x_k)_i}{\eta_k}, (d_k)_i\right\} = 0 = (\bar{d})_i.$$

(2) $\bar{x}_i = u_i$. Then by (2.21) we have

$$(v_k)_i = \frac{\min\{u_i, (x_k + \eta_k d_k)_i\} - (x_k)_i}{\eta_k}$$

and recalling that whenever $\bar{x}_i = u_i$ it must be $\bar{d}_i \leq 0$, we distinguish two subcases:

(a) when $\bar{d}_i < 0$, then $(v_k)_i = \min\left\{\frac{u_i - (x_k)_i}{\eta_k}, (d_k)_i\right\} = (d_k)_i$;

(b) when $\bar{d}_i = 0$, then

$$\lim_{k\to\infty, k\in K} (v_k)_i = \lim_{k\to\infty, k\in K} \min\left\{\frac{u_i - (x_k)_i}{\eta_k}, (d_k)_i\right\} = 0 = (\bar{d})_i.$$

(3) $l_i < \bar{x}_i < u_i$. Then by (2.20) or (2.21) we have $(v_k)_i = (x_k + \eta_k d_k - x_k)_i/\eta_k = (d_k)_i$.

These imply that $\lim_{k\to\infty, k\in K} v_k = \bar{d}$, so that (ii) is proved. $\square$

Finally, we are now ready to prove the main convergence result for Algorithm DFN$_{simple}$. We highlight that according to the following proposition, every limit point

of the sequence of iterates $\{x_k\}$, produced by Algorithm DFN_{simple}, is a stationary point for problem (2.1).

PROPOSITION 2.7. *Let* $\{x_k\}$ *be the sequence produced by Algorithm* DFN_{simple}. *Let* $\bar{x}$ *be any limit point of* $\{x_k\}$ *and* K *be the subset of indices such that*

$$\lim_{k\to\infty, k\in K} x_k = \bar{x}.$$

If the subsequence $\{d_k\}_K$ *is dense in the unit sphere (see Definition* 1.3*), then* $\bar{x}$ *is Clarke–Jahn stationary for problem* (2.1).

Proof. We recall that by Definition 2.3 we consider the stationarity condition at $\bar{x}$:

$$f^\circ(\bar{x}; \bar{d}) = \limsup_{\substack{y \to \bar{x}, y \in X \\ t \downarrow 0, y + t\bar{d} \in X}} \frac{f(y + t\bar{d}) - f(y)}{t} \geq 0 \qquad \forall \bar{d} \in D(\bar{x}).$$

We proceed by contradiction and assume that a direction $\bar{d} \in D(\bar{x}) \cap S(0,1)$ exists such that

$$(2.22) \qquad f^\circ(\bar{x}; \bar{d}) = \limsup_{\substack{x_k \to \bar{x}, x_k \in X, \\ t \downarrow 0, x_k + t\bar{d} \in X}} \frac{f(x_k + t\bar{d}) - f(x_k)}{t} < 0.$$

By recalling the instructions of the projected continuous search, if the condition at Step 1 is satisfied, we have $\alpha_k > 0$ and

$$(2.23) \qquad f([x_k + (\alpha_k/\delta)d_k]_{[l,u]}) > f(x_k) - \gamma(\alpha_k/\delta)^2;$$

otherwise, we have

$$(2.24) \qquad f([x_k + \tilde{\alpha}_k d_k]_{[l,u]}) > f(x_k) - \gamma\tilde{\alpha}_k^2.$$

Now, for every index $k \in K$, let us set

$$\eta_k = \begin{cases} \alpha_k/\delta & \text{if (2.23) holds,} \\ \tilde{\alpha}_k & \text{if (2.24) holds,} \end{cases}$$

and let v_k be defined as in relation (2.14) of Lemma 2.6, that is,

$$v_k = \frac{[x_k + \eta_k d_k]_{[l,u]} - x_k}{\eta_k}.$$

The instructions of Algorithm DFN_{simple} and definition of η_k guarantee that $\eta_k > 0$ for all $k \in K$. Moreover, by Proposition 2.5,

$$(2.25) \qquad \lim_{k\to\infty} \eta_k = 0.$$

Further, by Definition 1.3, a subset $\bar{K} \subseteq K$ exists such that

$$(2.26) \qquad \lim_{k\to\infty, k\in\bar{K}} x_k = \bar{x},$$

$$(2.27) \qquad \lim_{k\to\infty, k\in\bar{K}} d_k = \bar{d}.$$

Hence, by (2.25), (2.26), and (2.27), the assumptions of Lemma 2.6 are satisfied. Then, from point (i) of Lemma 2.6, we have $v_k \neq 0$, for $k \in \bar{K}$ and sufficiently large, so that relations (2.23) and (2.24) can be equivalently expressed as

$$f(x_k + \eta_k v_k) > f(x_k) - \gamma \eta_k^2,$$

that is, recalling that $\eta_k > 0$,

$$(2.28) \qquad \frac{f(x_k + \eta_k v_k) - f(x_k)}{\eta_k} > -\gamma \eta_k,$$

for $k \in \bar{K}$ and sufficiently large.

Then we can write

$$\limsup_{\substack{x_k \to \bar{x}, \, x_k \in X \\ t \downarrow 0, \, x_k + t\bar{d} \in X}} \frac{f(x_k + t\bar{d}) - f(x_k)}{t} \geq \limsup_{k \to \infty, k \in \bar{K}} \frac{f(x_k + \eta_k \bar{d}) - f(x_k)}{\eta_k}$$

$$= \limsup_{k \to \infty, k \in \bar{K}} \frac{f(x_k + \eta_k \bar{d}) + f(x_k + \eta_k v_k) - f(x_k + \eta_k v_k) - f(x_k)}{\eta_k}$$

$$\geq \limsup_{k \to \infty, k \in \bar{K}} \frac{f(x_k + \eta_k v_k) - f(x_k)}{\eta_k} - L\|\bar{d} - v_k\|,$$

where L is the Lipschitz constant of f. By (2.28) and (ii) of Lemma 2.6 we get, from the latter relation,

$$\limsup_{\substack{x_k \to \bar{x}, \, x_k \in X \\ t \downarrow 0, \, x_k + t\bar{d} \in X}} \frac{f(x_k + t\bar{d}) - f(x_k)}{t} \geq 0,$$

which contradicts (2.22) and concludes the proof. □

We conclude this section by reporting the following corollary, where we assume that the whole sequence $\{x_k\}$ converges to a single limit point $\bar{x}$.

COROLLARY 2.8. *Let $\{x_k\}$ be the sequence produced by Algorithm DFN$_{simple}$. If the sequence $\{x_k\}$ admits a unique limit point $\bar{x}$ and the sequence $\{d_k\}$ is dense in the unit sphere (see Definition 1.3), then $\bar{x}$ is Clarke–Jahn stationary for problem (2.1).*

Proof. The proof easily follows from Proposition 2.7 by considering $K = \{0, 1, \ldots\}$. □

2.2. Combining DFN$_{simple}$ with coordinate searches. A possible way to improve the efficiency of Algorithm DFN$_{simple}$ is to take advantage of the experience in the smooth case. For example, we can draw inspiration from the paper [34], where the objective function is repeatedly investigated along the directions $\pm e^1, \ldots, \pm e^n$ in order to capture the local behavior of the objective function. In fact, the use of a set of search directions, which is constant with iterations, allows us to store the actual and tentative steplengths, i.e., α^i and $\tilde{\alpha}^i$, respectively, that roughly summarize the sensitivity of the function along those directions. Thus, when the function is further investigated along such search directions, we can exploit information gathered in the previous searches along them.

In the following, we propose a new algorithm, where we first explore the coordinate directions and then, provided that the steplengths α^i and $\tilde{\alpha}^i$ are smaller than a given

threshold $\eta > 0$, a further direction d_k is explored. In particular, the sampling along the coordinate directions is performed by means of a continuous search procedure [34, 31].

ALGORITHM CS-DFN.

Input. $\theta \in (0,1)$, $\eta > 0$, $x_0 \in X$, $\tilde{\alpha}_0 > 0$, $\tilde{\alpha}_0^i > 0$, $d_0^i = e^i$, for $i = 1, \ldots, n$, a sequence $\{d_k\}$ of search directions such that $\|d_k\| = 1$, for all k.

For $k = 0, 1, \ldots$
    Set $y_k^1 = x_k$.
    For $i = 1, \ldots, n$
        Compute α and d_{k+1}^i by the continuous search $(\tilde{\alpha}_k^i, y_k^i, d_k^i; \alpha, d_{k+1}^i)$.

        If $(\alpha = 0)$ **then** set $\alpha_k^i = 0$ and $\tilde{\alpha}_{k+1}^i = \theta \tilde{\alpha}_k^i$
        else set $\alpha_k^i = \alpha$ and $\tilde{\alpha}_{k+1}^i = \alpha$.

        Set $y_k^{i+1} = y_k^i + \alpha_k^i d_{k+1}^i$.

    End For
    If $\left(\max_{i=1,\ldots,n}\{\alpha_k^i, \tilde{\alpha}_k^i\} \leq \eta \right)$ **then**
        Compute α_k and $\tilde{d}_k$ by the
            projected continuous search $(\tilde{\alpha}_k, y_k^{n+1}, d_k; \alpha_k, \tilde{d}_k)$.
        If $(\alpha_k = 0)$ **then** $\tilde{\alpha}_{k+1} = \theta \tilde{\alpha}_k$ and $y_k^{n+2} = y_k^{n+1}$
        else $\tilde{\alpha}_{k+1} = \alpha_k$ and $y_k^{n+2} = [y_k^{n+1} + \alpha_k \tilde{d}_k]_{[l,u]}$.
    else set $\tilde{\alpha}_{k+1} = \tilde{\alpha}_k$ and $y_k^{n+2} = y_k^{n+1}$.

    Find $x_{k+1} \in X$ such that $f(x_{k+1}) \leq f(y_k^{n+2})$.
End For
Output. The sequences $\{x_k\}$, $\{\alpha_k\}$, $\{\tilde{\alpha}_k\}$, $\{\alpha_k^i\}$ and $\{\tilde{\alpha}_k^i\}$, for $i = 1, \ldots, n$.

CONTINUOUS SEARCH $(\tilde{\alpha}, y, p; \alpha, p^+)$.
Data. $\gamma > 0$, $\delta \in (0,1)$.
Step 1. Compute the largest $\bar{\alpha}$ such that $y + \bar{\alpha}p \in X$. Set $\alpha = \min\{\bar{\alpha}, \tilde{\alpha}\}$.
Step 2. If $\alpha > 0$ and $f(y + \alpha p) \leq f(y) - \gamma\alpha^2$ **then** set $p^+ = p$ and go to Step 6.
Step 3. Compute the largest $\bar{\alpha}$ such that $y - \bar{\alpha}p \in X$. Set $\alpha = \min\{\bar{\alpha}, \tilde{\alpha}\}$.
Step 4. If $\alpha > 0$ and $f(y - \alpha p) \leq f(y) - \gamma\alpha^2$ **then** set $p^+ = -p$ and go to Step 6.
Step 5. Set $\alpha = 0$, **return** α and $p^+ = p$.
Step 6. Let $\beta = \min\{\bar{\alpha}, (\alpha/\delta)\}$.
Step 7. If $\alpha = \bar{\alpha}$ or $f(y + \beta p^+) > f(y) - \gamma\beta^2$ **return** α, p^+.
Step 8. Set $\alpha = \beta$ and go to Step 6.

Concerning the above definition of Algorithm CS-DFN, we again remark that the lack of a stopping condition allows us to study the asymptotic convergence properties of CS-DFN.

The following three propositions concern the convergence analysis of Algorithm CS-DFN. The third proof is omitted since it is very similar to the corresponding one for Algorithm DFN$_{simple}$.

PROPOSITION 2.9. *The continuous search cannot infinitely cycle between Step* 6 *and Step* 8.

Proof. We proceed by contradiction and assume that an infinite monotonically increasing sequence of positive numbers $\{\beta_j\}$ exists such that

$$\beta_j < \bar{\alpha} \quad \text{and} \quad f(y + \beta_j p^+) \leq f(y) - \gamma\beta_j^2.$$

The above relation contradicts the fact that X is compact, by definition, and that function f in problem (2.1) is continuous. □

The proposition that follows concerns convergence to zero of the steplengths in Algorithm CS-DFN. In particular, since α_k^i and $\tilde{\alpha}_k^i$ tend to zero, it results that the search along the dense direction d_k is performed eventually infinitely many times.

PROPOSITION 2.10. *Let $\{\alpha_k^i\}$, $\{\tilde{\alpha}_k^i\}$, $\{\alpha_k\}$, and $\{\tilde{\alpha}_k\}$ be the sequences produced by Algorithm CS-DFN; then*

$$(2.29) \qquad \lim_{k \to \infty} \max\{\alpha_k^1, \tilde{\alpha}_k^1, \ldots, \alpha_k^n, \tilde{\alpha}_k^n\} = 0,$$

$$(2.30) \qquad \lim_{k \to \infty} \max\{\alpha_k, \tilde{\alpha}_k\} = 0.$$

Proof. Reasoning as in the proof of Proposition 1 in [34], we can prove (2.29).

Now we have to show (2.30). By virtue of (2.29), we know that an index $\bar{k}$ exists such that the dense direction d_k is investigated for all $k \geq \bar{k}$.

Then, without loss of generality, we split the iteration sequence $\{k : k \geq \bar{k}\}$ into two sets K_1 and K_2, with $K_1 \cup K_2 = \{k\}$ and $K_1 \cap K_2 = \emptyset$. We denote

- K_1 the set of iterations when $\tilde{\alpha}_{k+1} = \alpha_k$,
- K_2 the set of iterations when $\tilde{\alpha}_{k+1} = \theta\tilde{\alpha}_k$.

Hence, the proof follows by reasoning as in the proof of Proposition 2.5. □

PROPOSITION 2.11. *Let $\{x_k\}$ be the sequence produced by Algorithm CS-DFN. Let $\bar{x}$ be any limit point of $\{x_k\}$ and K be the subset of indices such that*

$$\lim_{k \to \infty, k \in K} x_k = \bar{x}.$$

If the subsequence $\{d_k\}_K$ is dense in the unit sphere (see Definition 1.3), then $\bar{x}$ is Clarke–Jahn stationary for problem (2.1).

Proof. The proof trivially follows from Proposition 2.7. □

3. The nonsmooth nonlinearly constrained case. In this section, we consider Lipschitz-countinuous nonlinearly constrained problems of the following form:

$$(3.1) \qquad \begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) \leq 0, \\ & l \leq x \leq u, \end{aligned}$$

where $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}^m$, and $l, u \in \mathbb{R}^n$. The vectors l and u correspond respectively to lower and upper bounds on the variables $x \in \mathbb{R}^n$ and satisfy the additional condition $l < u$. We also assume throughout the paper that $f(x)$ and $g(x)$ are Lipschitz continuous functions, though they may be possibly nondifferentiable. Furthermore, $\mathcal{F}$ indicates the feasible set of problem (3.1), i.e.,

$$\mathcal{F} = \{x \in X : g(x) \leq 0\}.$$

We highlight that, by definition, $X = \{x \in \mathbb{R}^n : l \leq x \leq u\}$ is a compact subset of $\mathbb{R}^n$.

3.1. Preliminary results. The nonlinearly constrained problem (3.1) can be handled partitioning the constraints in two different sets, the first one defined by general inequality constraints, and the second one consisting of simple bound constraints. Then, for this kind of problem, we can state necessary optimality conditions that explicitly take into account the presence of these two different sets of constraints.

The following propositions extend the results in [21, Theorem 6] to the case where inequality constraints and an additional convex set of constraints are present.

PROPOSITION 3.1 (Fritz John optimality conditions). *Let $x^\star \in \mathcal{F}$ be a local minimum of problem* (3.1). *Then, multipliers $\lambda_0^\star, \lambda_1^\star, \ldots, \lambda_m^\star \in \mathbb{R}$ not all zero exist, with*

$$\lambda_0^\star \geq 0, \quad \lambda_i^\star \geq 0, \quad and \quad \lambda_i^\star g_i(x^\star) = 0 \ \forall \ i = 1, \ldots, m,$$

such that for every $d \in D(x^\star)$

$$(3.2) \qquad \max\left\{\xi^\top d : \xi \in \lambda_0^\star \partial f(x^\star) + \sum_{i=1}^m \lambda_i^\star \partial g_i(x^\star)\right\} \geq 0.$$

Proof. The proof can be found in Appendix A. □

As usual, by adding a version of the Mangasarian–Fromowitz constraint qualification condition for nonsmooth problems, we can prove KKT necessary optimality conditions.

COROLLARY 3.2 (KKT necessary optimality conditions). *Let $x^\star \in \mathcal{F}$ be a local minimum of problem* (3.1) *and assume that a direction $d \in D(x^\star)$ exists such that for all $i \in \{1, \ldots, m : g_i(x^\star) = 0\}$,*

$$(3.3) \qquad (\xi^{g_i})^\top d < 0 \quad \forall \ \xi^{g_i} \in \partial g_i(x^\star).$$

Then, multipliers $\lambda_1^\star, \ldots, \lambda_m^\star \in \mathbb{R}$ exist, with

$$\lambda_i^\star \geq 0 \quad and \quad \lambda_i^\star g_i(x^\star) = 0, \ \forall \ i = 1, \ldots, m,$$

such that for every $d \in D(x^\star)$

$$\max\left\{\xi^\top d : \xi \in \partial f(x^\star) + \sum_{i=1}^m \lambda_i^\star \partial g_i(x^\star)\right\} \geq 0.$$

Proof. The proof can be found in Appendix A. □

As regards the stationarity conditions for problem (3.1), taking into account the above results, we can now give the following definition.

DEFINITION 3.3 (stationary point). *Given problem* (3.1), *the feasible point $\bar{x}$ is a stationary point of* (3.1) *if multipliers $\bar{\lambda}_1, \ldots, \bar{\lambda}_m \in \mathbb{R}$ exist, with*

$$\bar{\lambda}_i \geq 0 \quad and \quad \bar{\lambda}_i g_i(\bar{x}) = 0 \ \forall \ i = 1, \ldots, m,$$

such that for every $d \in D(\bar{x})$

$$\max \left\{ \xi^\top d : \xi \in \partial f(\bar{x}) + \sum_{i=1}^m \bar{\lambda}_i \partial g_i(\bar{x}) \right\} \geq 0.$$

3.2. The penalty approach. Given problem (3.1), we introduce the penalty function

$$Z_\varepsilon(x) = f(x) + \frac{1}{\varepsilon} \sum_{i=1}^m \max \{0, g_i(x)\}$$

and define the penalized problem

(3.4)
$$\begin{aligned} \min \quad & Z_\varepsilon(x) \\ \text{s.t.} \quad & x \in X. \end{aligned}$$

Remark 3.4. Observe that since f and g_i, $i = 1, \ldots, m$, are Lipschitz continuous, with Lipschitz constants L_f and L_{g_i}, $i = 1, \ldots, m$, the penalty function Z_ε is Lipschitz continuous too, with Lipschitz constant

$$L \leq L_f + \frac{1}{\varepsilon} \sum_{i=1}^m L_{g_i}.$$

Remark 3.5. Note that problem (3.4), for any $\varepsilon > 0$, has the same structure and properties of problem (2.1).

We further note that our penalty approach differs from the ones previously proposed in the literature (see, e.g., [15] and references therein), since only the general nonlinear constraints are penalized. The minimization of the penalty function is then carried out on the set defined by the bound constraints. We report in the following proposition the equivalence between problem (3.4) and the nonlinearly constrained problem (3.1).

In order to carry out the theoretical analysis, we use an extended version of the Mangasarian–Fromowitz constraint qualification (EMFCQ) condition for nonsmooth problems.

ASSUMPTION 1 (EMFCQ). *Given problem* (3.1), *for any $x \in X \backslash \overset{\circ}{\mathcal{F}}$ a direction $d \in D(x)$ exists such that*

$$(\xi^{g_i})^\top d < 0$$

for all $\xi^{g_i} \in \partial g_i(x)$, $i \in \{1, \ldots, m : g_i(x) \geq 0\}$.

PROPOSITION 3.6. *Let Assumption 1 hold. Given problem* (3.1) *and considering problem* (3.4), *a threshold value $\varepsilon^\star > 0$ exists such that for every $\varepsilon \in (0, \varepsilon^\star]$, every Clarke–Jahn stationary point $\bar{x}$ of problem* (3.4) *is stationary (according to Definition 3.3) for problem* (3.1).

Proof. The proof is reported in Appendix B. $\quad\square$

3.3. A derivative-free algorithm. Now we report the algorithm adopted for solving problem (3.4), which is obtained from Algorithm CS-DFN by replacing f with Z_ε for given $\varepsilon > 0$. For simplicity, we omit reporting the extension of Algorithm

DFN$_{simple}$ to the general inequality constrained case, which requires trivial modifications.

ALGORITHM DFN$_{con}$.

Input. $\theta \in (0,1)$, $x_0 \in X$, $\varepsilon > 0$, $\tilde{\alpha}_0 > 0$, $\tilde{\alpha}_0^i > 0$, $d_0^i = e^i$, for $i = 1, \ldots, n$, a sequence $\{d_k\}$ such that $\|d_k\| = 1$, for all k.

For $k = 0, 1, \ldots$
    Set $y_k^1 = x_k$.
    For $i = 1, \ldots, n$
        Compute α and d_{k+1}^i by the continuous search $(\tilde{\alpha}_k^i, y_k^i, d_k^i; \alpha, d_{k+1}^i)$.

        If $(\alpha = 0)$ **then** set $\alpha_k^i = 0$ and $\tilde{\alpha}_{k+1}^i = \theta\tilde{\alpha}_k^i$
        else set $\alpha_k^i = \alpha$ and $\tilde{\alpha}_{k+1}^i = \alpha$.

        Set $y_k^{i+1} = y_k^i + \alpha_k^i d_{k+1}^i$.

    End For
    If $\big(\max_{i=1,\ldots,n} \{\alpha_k^i, \tilde{\alpha}_k^i\} \leq \eta \big)$ **then**
        Compute α_k and $\tilde{d}_k$ by the
            projected continuous search $(\tilde{\alpha}_k, y_k^{n+1}, d_k; \alpha_k, \tilde{d}_k)$.
        If $(\alpha_k = 0)$ **then** $\tilde{\alpha}_{k+1} = \theta\tilde{\alpha}_k$ and $y_k^{n+2} = y_k^{n+1}$
        else $\tilde{\alpha}_{k+1} = \alpha_k$ and $y_k^{n+2} = [y_k^{n+1} + \alpha_k\tilde{d}_k]_{[l,u]}$.
    else set $\tilde{\alpha}_{k+1} = \tilde{\alpha}_k$, $y_k^{n+2} = y_k^{n+1}$.
    Find $x_{k+1} \in X$ such that $Z_\varepsilon(x_{k+1}) \leq Z_\varepsilon(y_k^{n+2})$.
End For
Output. The sequences $\{x_k\}$, $\{\alpha_k\}$, $\{\tilde{\alpha}_k\}$, $\{\alpha_k^i\}$ and $\{\tilde{\alpha}_k^i\}$, for $i = 1, \ldots, n$.

We remark that in Algorithm DFN$_{con}$ the continuous search procedure is performed replacing f with Z_ε. Further, observe that Algorithm DFN$_{con}$ can be used to solve the constrained problem (3.1) provided that the penalty parameter ε is sufficiently small, as the following proposition states.

PROPOSITION 3.7. *Let Assumption* 1 *hold and let* $\{x_k\}$ *be the sequence produced by Algorithm DFN$_{con}$. Let* $\bar{x}$ *be any limit point of* $\{x_k\}$ *and* K *be the subset of indices such that*

$$\lim_{k \to \infty, k \in K} x_k = \bar{x}.$$

If the subsequence $\{d_k\}_K$ *is dense in the unit sphere (see Definition* 1.3*), then a threshold value* ε^* *exists such that for all* $\varepsilon \in (0, \varepsilon^*]$, $\bar{x}$ *is stationary for problem* (3.1).

Proof. The proof follows from Propositions 2.11 and 3.6. ☐

4. Implementation details and numerical results. This section investigates the numerical issues related to the implementation of the proposed algorithms. We first report the numerical experience related to bound-constrained problems; then we analyze the computational results related to the nonlinearly constrained case. In order to compare algorithms using data and performance profiles, all the experiments have been conducted allowing for a maximum number of 20,000 function evaluations.

For the parameters included in the proposed algorithms (DFN_{simple}, CS-DFN, DFN_{con}) we considered the following setting: $\theta = 0.5$, $\gamma = 10^{-6}$, $\delta = 0.5$, $\eta = 10^{-3}$,

$$\tilde{\alpha}_0^i = \max\left\{10^{-3}, \min\{1, |(x_0)_i|\}\right\}, \qquad i = 1, \ldots, n,$$

$$\tilde{\alpha}_0 = \frac{1}{n} \sum_{i=1}^{n} \tilde{\alpha}_0^i.$$

Regarding the choice of the new iterate x_{k+1}, we remark that

- in Algorithm DFN_{simple}, x_{k+1} is computed starting from $\tilde{x}_k$ and performing projected continuous searches along a set of $n - 1$ directions which define an orthonormal basis in $\mathbb{R}^n$ along with d_k;
- in Algorithms CS-DFN and DFN_{con}, if $(\max_{i=1,\ldots,n}\{\alpha_k^i, \tilde{\alpha}_k^i\} \leq \eta)$, then x_{k+1} is computed as above but starting from y_k^{n+2}. Otherwise, we set $x_{k+1} = y_k^{n+2}$.

As a final note, by drawing inspiration from [26, Theorem 6.4] and from the proof of Proposition 2.7, and by recalling that by Proposition 2.5 $\lim_{k\to\infty} \max\{\alpha_k, \tilde{\alpha}_k\} = 0$, in the implementation of our algorithms, we used as a termination condition the following heuristic rule:

$$(4.1) \qquad\qquad \max\{\alpha_k, \tilde{\alpha}_k\} \leq 10^{-13}.$$

However, we highlight that the algorithms are compared by means of performance and data profiles [37], that is, by using a normalized convergence test on the function values. Thus, we adopted the tight convergence test (4.1) in order to provide enough information on the progress of all the codes compared.

The codes DFN_{simple}, CS-DFN, and DFN_{con} are freely available for download at http://www.dis.uniroma1.it/~lucidi/DFL.

4.1. Bound-constrained problems. The first part of the numerical experience has been carried out on a set of 142 bound-constrained nonsmooth problems from [45], [32], and [37], with a number of variables n in the range $[1, 200]$ (see Table 1).

As showed in the theoretical analysis of the different algorithms, our linesearch-based approach is able to guarantee convergence toward stationary points of the non-smooth problem, provided that suitable sequences of search directions $\{d_k\}$ are dense in the unit sphere (see Definition 1.3). In particular, we can adopt the mapping based on the Halton sequence [19], which is the one implemented in the NOMAD package [1, 2, 27]. But, unlike NOMAD, further mappings can be used in our algorithms to generate the sequence of search directions $\{d_k\}$, since we are not committed to using a modified Halton sequence in order to generate points on a mesh (see, e.g., [19]). For instance, we implemented a mapping based on the Sobol sequence [43, 8], which is a pseudorandom generator widely used in practice.

In order to show the behavior of the above pseudorandom sequences, we preliminarily compared two versions of the Algorithm DFL_{simple}, which respectively use

TABLE 1
Distribution of problem dimensions for the bound-constrained case.

n	1	2	3	4	5	6	7	8	9	10
Number of problems	4	34	12	17	10	8	6	6	6	10

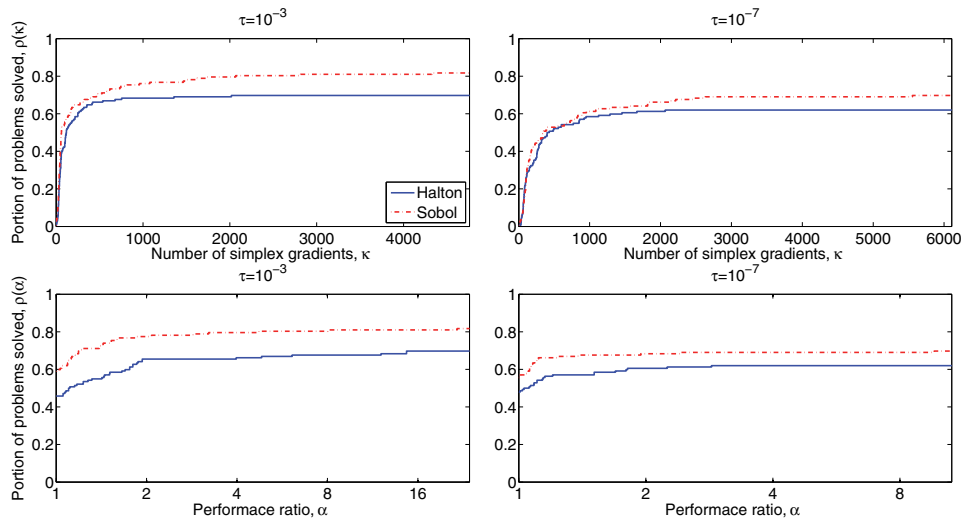| n	11	12	15	20	50	100	200			
Number of problems	6	6	1	8	4	2	2			

FIG. 1. *Data (top) and performance (bottom) profiles for the* 142 *bound-constrained problems. Comparison between Sobol and Halton pseudorandom sequences within* DFN_{simple}.

the Halton and the Sobol sequence, on the test set of bound-constrained problems described above. The resulting experience is reported in Figure 1 using data and performance profiles [37].

We recall that for a given value of the tolerance τ, data profiles report the percentage of problems which are solved (by each solver) within κ simplex gradient evaluations ($n + 1$ being the number of function evaluations necessary for each simplex gradient computation). On the other hand, for each solver, a performance profile reports the percentage of problems which are solved within α-times the function evaluations required by the best solver. Hence, the uppermost curve in the profiles denotes better performances of the corresponding algorithm.

As we can see, the Sobol pseudorandom sequence outperforms the Halton one for all precision levels, in terms of both efficiency and robustness. Then, we compared both Algorithms DFN_{simple} and CS-DFN and report the results in Figure 2 in terms of performance and data profiles.

As we can see, the combination of coordinate and dense directions can improve the performance of the algorithm.

Finally, we compared CS-DFN with two state-of-the-art derivative-free optimization codes, namely, NOMAD [1, 27, 2] and BOBYQA [40]. We ran NOMAD by using its default settings and BOBYQA by specifying `RHOBEG` = 1 and `RHOEND` = 10^{-13}. The results of this comparison are summarized in Figure 3. By looking at how rapidly data and performance profiles rise for small values of κ and α, respectively, we can say that: (i) BOBYQA is very efficient for small precision; (ii) when a high precision, i.e., $\tau = 10^7$, is required, CS-DFN is the most efficient in terms of number of simplex gradient evaluations, whereas NOMAD is the best one in terms of performance ratio. As concerns the robustness of the compared solvers, which is represented by the asymptotic behavior of the reported data and performance profiles, the robustness of CS-DFN is between NOMAD, which is the most robust one, and BOBYQA. The above comments about efficiency and robustness of the methods were definitely expected and can be explained by the following considerations:
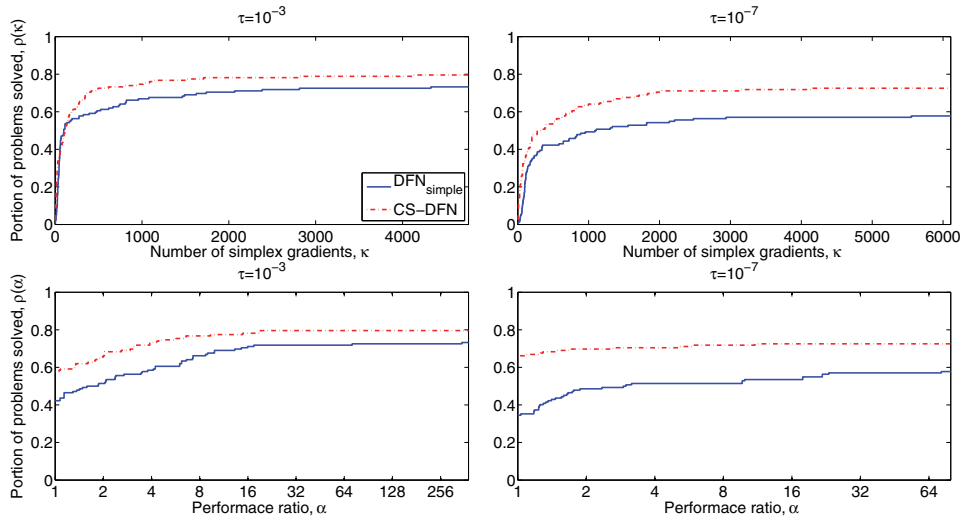
FIG. 2. *Data (top) and performance (bottom) profiles for the* 142 *bound-constrained problems. Comparison between* DFN_{simple} *and CS-DFN.*
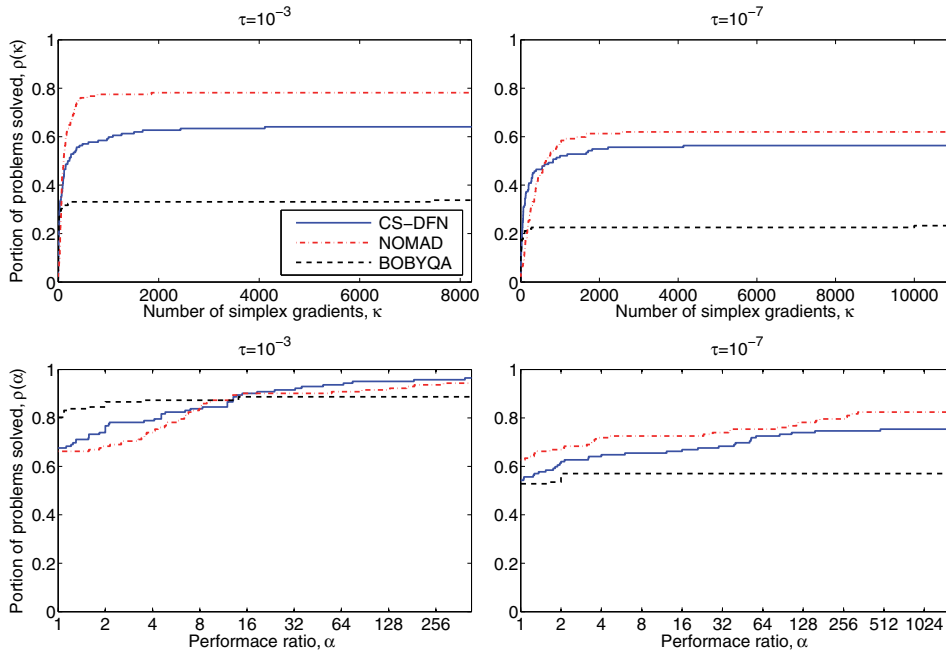

FIG. 3. *Data (top) and performance (bottom) profiles for the* 142 *bound-constrained problems. Comparison of CS-DFN, NOMAD, and BOBYQA.*

(i) BOBYQA is quite efficient, though not so effective, because it is a model-based method whose performances are strongly related to the smoothness of the objective function. It is worth noting that for smooth problems the performances of BOBYQA tend to improve sensibly.
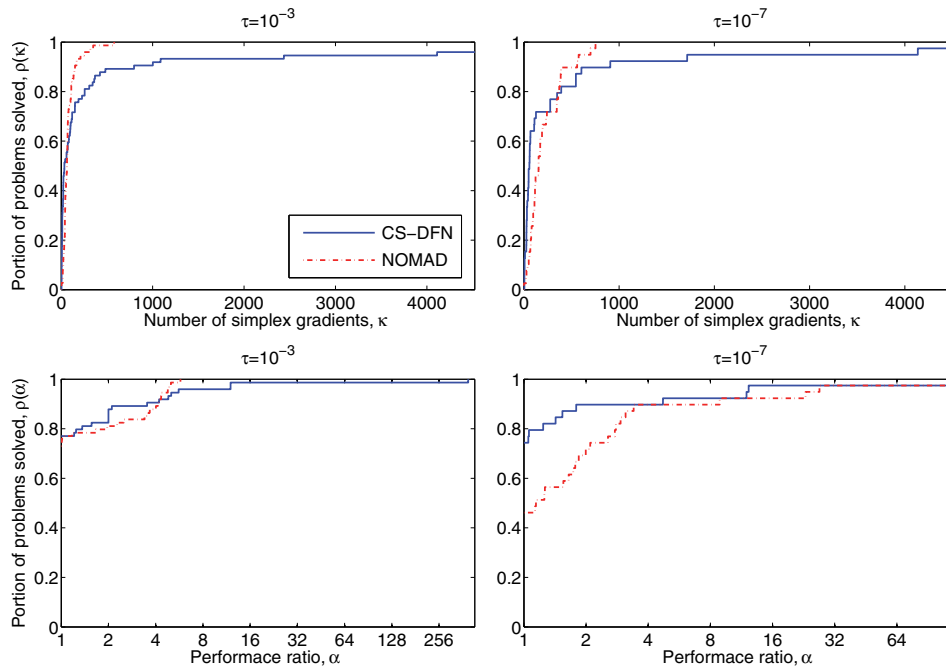
FIG. 4. *Data (top) and performance (bottom) profiles for CS-DFN and NOMAD on the set of problems where they find the same solution.*

(ii) Both CS-DFN and NOMAD are more robust than BOBYQA since they use globally convergent strategies which do not assume any continuous differentiability. For this reason, they are less efficient than BOBYQA and (potentially) more expensive from a computational point of view.

(iii) NOMAD is the most robust code because it incorporates a heuristic search phase (as opposed to the poll phase) in which quadratic models are used to try to improve the current iterate. This phase can surely help improving the quality of the solution, especially for nonconvex problems.

To better understand the behaviors of CS-DFN and NOMAD, we now limit the comparison to those problems where both methods find the same solution. More precisely, given a problem and the solution points $x^{*,1}$ and $x^{*,2}$ returned by the two solvers, the solutions are considered the same if

$$\frac{|f(x^{*,1}) - f(x^{*,2})|}{f(x_0) - \min\{f(x^{*,1}), f(x^{*,2})\}} \leq \tau$$

for a given precision $\tau > 0$. These results are reported in Figure 4, where it can be noted that the gap between CS-DFN and NOMAD in terms of robustness is considerably reduced. Figure 4 also shows the good behavior of the linesearch strategy of CS-DFN in terms of efficiency. Since the results in Figure 4 report convergence to the same stationary point, we claim that, for these problems, the search phase of NOMAD possibly does not help to improve efficiency.

These latter results motivate us to better investigate the behavior of the codes. We report the results of a further comparison between CS-DFN and NOMAD, where we
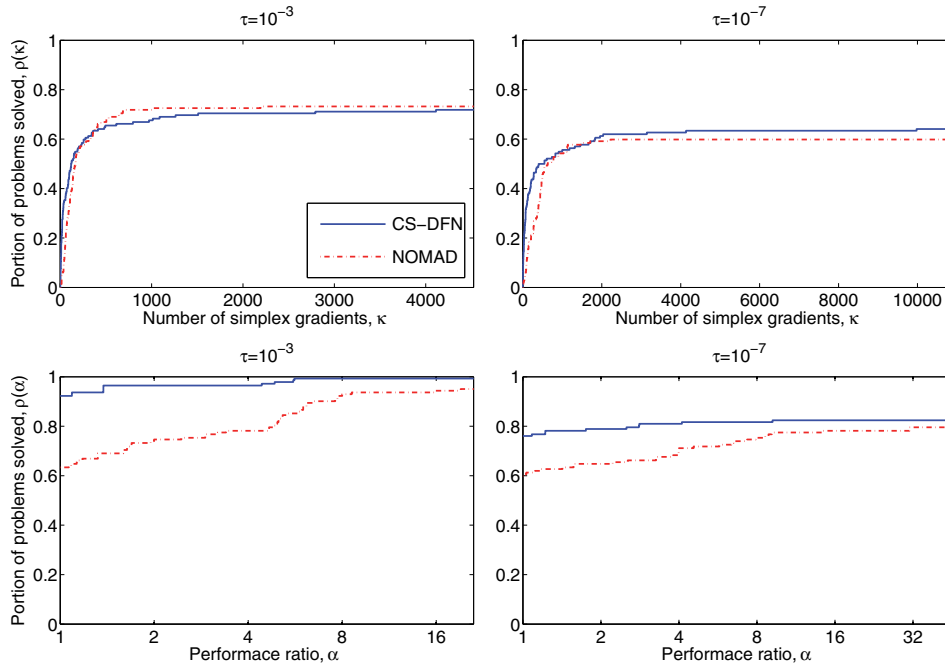
FIG. 5. *Data (top) and performance (bottom) profiles for the* 142 *bound-constrained problems. Comparison of CS-DFN and NOMAD (without quadratic local models).*

run NOMAD disabling the search phase with quadratic models by setting MODEL_SEARCH to NO, again on the whole set of 142 bound constrained problems.

The results reported in Figure 5 suggest that when NOMAD does not exploit the model search phase, to a large extent CS-DFN and NOMAD show similar performance in terms of robustness (with a slight preference for CS-DFN). On the other hand, it can be noted that the globalization strategy of CS-DFN, based on the use of linesearches, outperforms the strategy of NOMAD, based on the use of a MADS, in terms of efficiency.

4.2. Nonlinearly constrained problems. In the second part of our numerical experience, we defined a set of hard nonsmooth nonlinearly constrained test problems by pairing the objective functions of the collection [32] with the constraint families proposed in [23], thus obtaining 296 problems. The problems in this collection have a number of constraints m in the range $[1,199]$ and a number of variables n in the range $[1,200]$ (see Table 2). We note that 205 of 296 problems have a starting point x_0, which is not feasible, that is,

$$h(x_0) > 10^{-6} \qquad \text{with} \quad h(x) = \max\left\{0, \max_{i=1,\dots,m}\{g_i(x)\}\right\}.$$

In order to adapt the procedure for constructing performance and data profiles, as proposed in [37], to the nonlinearly constrained case, we considered the convergence test

$$\tilde{f}_0 - f(x) \geq (1 - \tau)(\tilde{f}_0 - f_L),$$

TABLE 2
Distribution of problem dimensions (n number of variables, m number of constraints) for the nonlinearly constrained test set.

n	2	3	4	5	6	10	20	50	100	200
Number of problems	96	30	40	10	10	20	40	10	20	20

m	1	2	3	4	5	6	8	9	11	12
Number of problems	151	39	17	24	9	1	3	2	3	2
m	18	19	22	23	48	49	98	99	198	199
Number of problems	9	6	3	2	3	2	6	4	6	4

where $\tilde{f}_0$ is the objective function value of the *worst feasible point* determined by all the solvers (note that in the bound-constrained case, $\tilde{f}_0 = f(x_0)$), $\tau > 0$ is a tolerance, and f_L is computed for each problem as the smallest value of f (at a feasible point) obtained by any solver within 20,000 function evaluations. We notice that when a point is not feasible (i.e., $h(x) > 10^{-6}$) we set $f(x) = +\infty$.

As concerns the penalty parameter ε that defines Algorithm DFN$_{con}$, we first tried different fixed values for this parameter, namely, $10^{-1}, 10^{-3}, 10^{-5}$. Then, we tried a more sophisticated managing and updating strategy. In particular, we used a vector of penalty parameters $\varepsilon \in \mathbb{R}^m$ and considered the penalty function

$$Z_\varepsilon(x) = f(x) + \sum_{i=1}^m \frac{1}{\varepsilon_i} \max\{0, g_i(x)\},$$

which trivially preserves all the theoretical results proved in section 3. The vector of penalty parameters is iteratively updated during progress of the algorithm and, in particular, we chose

$$(\varepsilon_0)_i = \begin{cases} 10^{-3} & \text{if } \max\{0, g_i(x_0)\} < 1, \\ 10^{-1} & \text{otherwise,} \end{cases} \qquad i = 1, \ldots, m,$$

and adopted the updating rule

$$(\varepsilon_{k+1})_i = \begin{cases} 10^{-2}(\varepsilon_k)_i & \text{if } (\varepsilon_k)_i g_i(x_k) > \max\{\alpha_k, \tilde{\alpha}_k\}, \\ (\varepsilon_k)_i & \text{otherwise,} \end{cases} \qquad i = 1, \ldots, m.$$

The above updating rule is applied right before computation of the new iterate x_{k+1}. We notice that the rule described above takes inspiration from derivative-based exact penalty approaches (see, e.g., [30], [38]), where the updating rule for the penalty parameter is based on the (scaled) comparison between the stationarity measure of the point and the constraint violation. In a derivative-free context, the stationarity measure can be approximated by means of the steplengths selected along the search directions, as shown in [26].

First we compare the different versions of DFN$_{con}$ with the above described strategies for the parameter ε. The results of this comparison are reported in Figure 6,
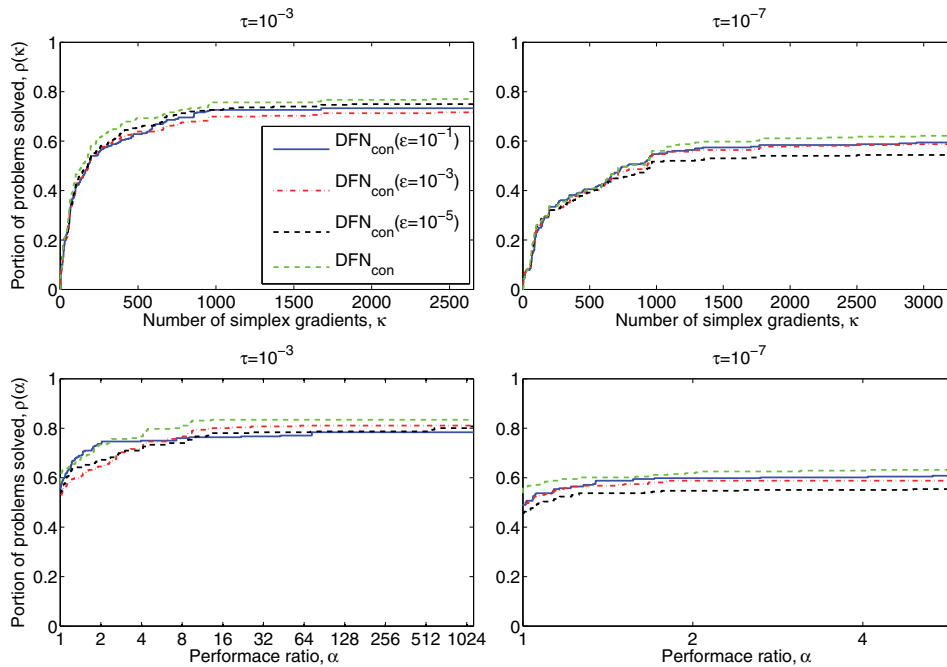
FIG. 6. *Data (top) and performance (bottom) profiles for the* 296 *constrained problems. Comparison of different versions of* DFN_{con} *with different strategies for updating the penalty parameter* ε.

from which it emerges that, though the performances of the algorithms are quite similar to each other, the scheme where parameter ε is adaptively updated looks preferable.

Then, in Figure 7, we report the comparison among DFN_{con}, NOMAD, and COBYLA [39]. NOMAD was run by setting the constraints type to PEB [7], so that constraints are treated first with the progressive barrier and, once satisfied, with the extreme barrier approach. COBYLA was run by setting RHOBEG $= 1$ and RHOEND $= 10^{-13}$. As already said, a maximum number of 20,000 function evaluations was specified for all the solvers. As can be seen, when relatively low precision is required, COBYLA has an initial fast progress but is not as robust as the other two codes. This is mainly due to the nonsmoothness of the problems which likely yields convergence to nonstationary points. Indeed, as already said for BOBYQA, for smooth problems the performances of COBYLA tend to improve sensibly. When high precision is required, NOMAD is the most robust solver and DFN_{con} is slightly more efficient with respect to the data profiles.

Again, as done for the bound constrained case, to better understand the behavior of NOMAD and DFN_{con}, we now limit the comparison of the two codes to those problems where both the solvers find the same feasible solution. These results are reported in Figure 8. It emerges that the gap between NOMAD and DFN_{con} is considerably reduced. This seems to confirm that NOMAD has a greater ability to find better solutions with respect to those found by DFN_{con}.

Finally, we again compare DFN_{con} and NOMAD on the whole test set but by setting the parameter MODEL SEARCH to NO in NOMAD. The results are reported in

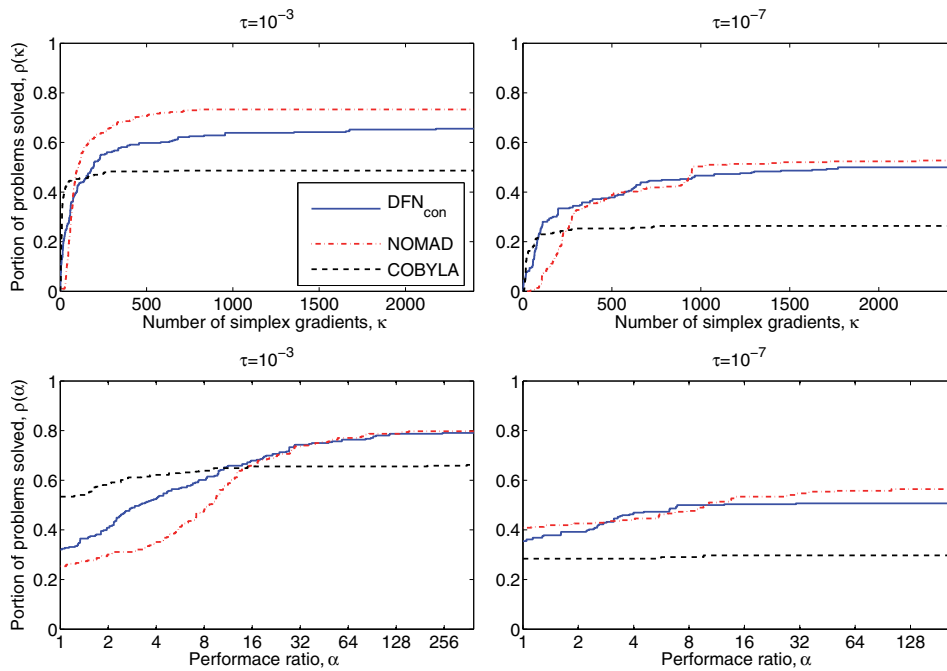FIG. 7. *Data (top) and performance (bottom) profiles for the* 296 *constrained problems. Comparison of DFN$_{con}$, NOMAD, and COBYLA.*
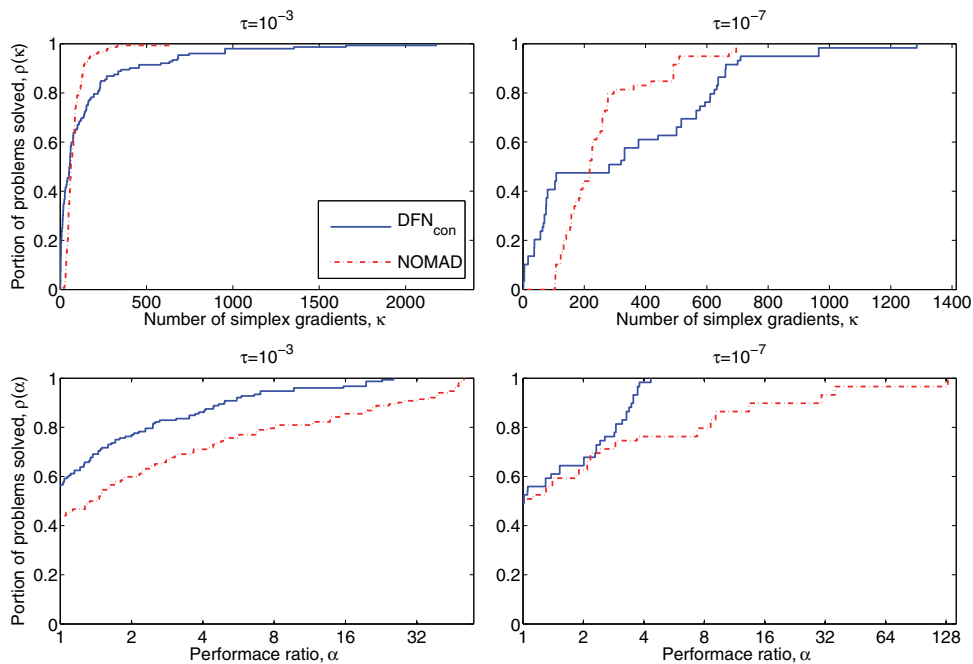

FIG. 8. *Data (top) and performance (bottom) profiles for DFN$_{con}$ and NOMAD on the set of problems where they find the same solution.*
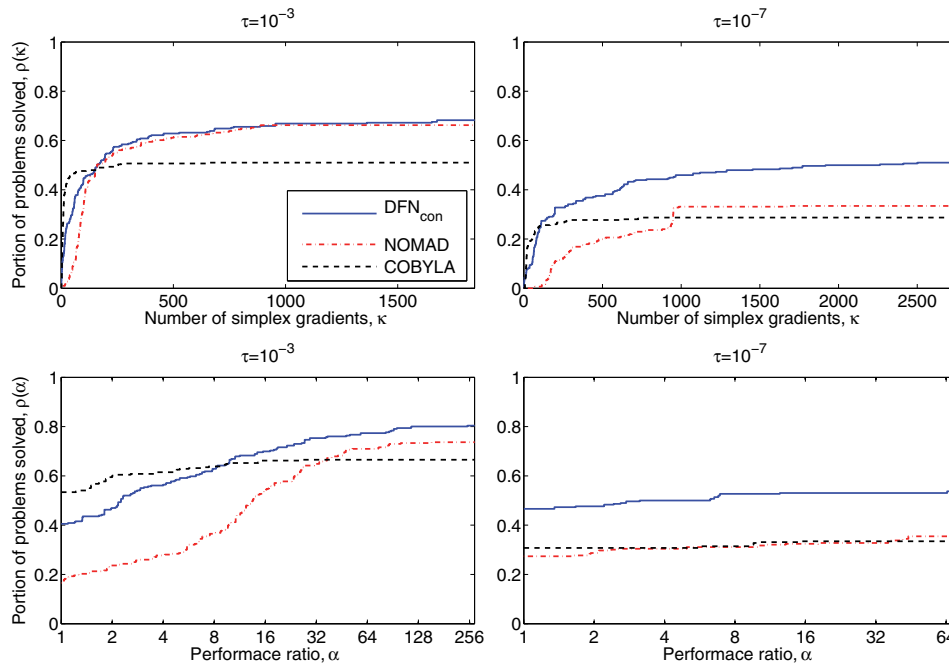
FIG. 9. *Data (top) and performance (bottom) profiles for the* 296 *constrained problems. Comparison of DFN$_{con}$, NOMAD (without quadratic models), and COBYLA.*

Figure 9 and confirm that the robustness of NOMAD was largely due to the use of quadratic models to heuristically improve the current iterate.

5. Conclusions. In this paper, we described new methods for dealing with nonsmooth optimization problems when no first-order information is available. We adopted a projected linesearch approach. In particular, we extended the linesearch approach with sufficient decrease for smooth minimization problems. This approach gives a twofold achievement. On the one hand, by means of the sufficient decrease we can avoid the use of integer lattices. On the other hand, the extrapolation phase allows us to better exploit a descent direction and hence to characterize all the limit points of the sequence of iterates, under some density assumptions on the search directions.

In the first part of the paper, we considered problems with only bound constraints on the variables and we proposed two different algorithms for their solution. We showed that every accumulation point of the sequence of iterates produced by both the algorithms is Clarke–Jahn stationary. As concerns nonlinear inequality constrained problems, we introduced the use of an exact penalty function to transform the given problem into a bound-constrained one, which is solved by adapting the method proposed for the bound-constrained case. Similarly to the bound constrained case, we were able to prove again that every accumulation point of the generated sequence of iterates is Clarke stationary for the original constrained problem.

The numerical results reported in the paper show that the use of linesearches gives a large freedom in the choice of the set of used search directions. Furthermore, our analysis highlights the fact that coordinate searches can often improve the

performance of the proposed algorithms. Finally, we compared the proposed methods with other state-of-the-art codes on two large test sets of bound-constrained and non-linearly constrained nonsmooth problems. The numerical experimentation carried out evidenced that our globalization strategy is promising, as compared to the MADS, and at the same time showed the importance of using approximating models to determine good solution points.

Appendix A. Necessary optimality conditions.

Proof of Proposition 3.1. By definition of local optimality of $x^\star$, we know that a constant $\rho > 0$ exists such that

$$(A.1) \qquad f(x^\star) \leq f(x) \qquad \forall\ x \in \mathcal{F} \cap \mathcal{B}(x^\star, \rho),$$

where $\mathcal{B}(x^\star, \rho) = \{x \in \mathbb{R}^n : \|x - x^\star\| \leq \rho\}$. Then, let us introduce the functional

$$\Phi(x) = \max \big\{ f(x) - f(x^\star), g_1(x), \ldots, g_m(x),$$
$$(l_1 - x_1), \ldots, (l_n - x_n), (x_1 - u_1), \ldots, (x_n - u_n) \big\}$$

and show that for all $x \in \mathcal{B}(x^\star, \rho)$, $\Phi(x) \geq 0$. Indeed, by contradiction, suppose that $\hat{x} \in \mathcal{B}(x^\star, \rho)$ exists such that $\Phi(\hat{x}) < 0$. This implies that $g(\hat{x}) < 0$ and $l < \hat{x} < u$, yielding $\hat{x} \in \mathcal{F}$. Further, $f(\hat{x}) < f(x^\star)$. This latter condition contradicts the optimality condition (A.1) of $x^\star$.

Now, since $x^\star \in \mathcal{F}$ and $\Phi(x^\star) = 0$, we know that $x^\star$ is a local minimum of $\Phi(x)$ onto $\mathbb{R}^n$. Hence, by definition of Clarke stationarity

$$0 \in \partial\Phi(x^\star).$$

Then, considering [9, Proposition 2.3.12], we have that

$$(A.2) \qquad 0 \in \tilde{\lambda}_0 \partial f(x^\star) + \sum_{i \in I_0(x^\star)} \tilde{\lambda}_i \partial g_i(x^\star) - \sum_{j \in I_l(x^\star)} \tilde{\mu}_j e_j + \sum_{h \in I_u(x^\star)} \tilde{\mu}_h e_h$$

with $I_0(x^\star) = \{i : g_i(x^\star) = 0\}$, $I_l(x^\star) = \{j : x_j^\star = l_j\}$, $I_u(x^\star) = \{h : x_h^\star = u_h\}$, $\tilde{\lambda}_0 \geq 0$, $\tilde{\lambda}_i \geq 0$, $i \in I_0(x^\star)$, $\tilde{\mu}_j \geq 0$, $j \in I_l(x^\star)$, $\tilde{\mu}_h \geq 0$, $h \in I_u(x^\star)$, and

$$(A.3) \qquad \tilde{\lambda}_0 + \sum_{i \in I_0(x^\star)} \tilde{\lambda}_i + \sum_{j \in I_l(x^\star)} \tilde{\mu}_j + \sum_{h \in I_u(x^\star)} \tilde{\mu}_h = 1.$$

Now, from the linear independence of the set $\{e_j, e_h, j \in I_l(x^\star), h \in I_u(x^\star)\}$, it turns out that

$$\tilde{\lambda}_0 + \sum_{i \in I_0(x^\star)} \tilde{\lambda}_i \neq 0.$$

Indeed, if this was not the case, relation (A.2) would yield

$$0 = - \sum_{j \in I_l(x^\star)} \tilde{\mu}_j e_j + \sum_{h \in I_u(x^\star)} \tilde{\mu}_h e_h,$$

which would then impose $\tilde{\mu}_j$'s and $\tilde{\mu}_h$'s to be all zero, thus contradicting (A.3).

Hence, by dividing (A.2) by $\tilde{\lambda}_0 + \sum_{i \in I_0(x^\star)} \tilde{\lambda}_i = \Lambda \neq 0$ and by posing

$$\lambda_0 = \frac{\tilde{\lambda}_0}{\Lambda},$$

$$\lambda_i = \frac{\tilde{\lambda}_i}{\Lambda} \ \forall \ i \in I_0(x^\star),$$

$$\mu_j = \frac{\tilde{\mu}_j}{\Lambda} \ \forall \ j \in I_l(x^\star),$$

$$\mu_h = \frac{\tilde{\mu}_h}{\Lambda} \ \forall \ h \in I_u(x^\star),$$

we get

$$(A.4) \qquad 0 \in \lambda_0 \partial f(x^\star) + \sum_{i \in I_0(x^\star)} \lambda_i \partial g_i(x^\star) - \sum_{j \in I_l(x^\star)} \mu_j e_j + \sum_{h \in I_u(x^\star)} \mu_h e_h$$

with $\lambda_0 \geq 0$, $\lambda_i \geq 0$, $i \in I_0(x^\star)$, $\mu_j \geq 0$, $j \in I_l(x^\star)$, $\mu_h \geq 0$, $h \in I_u(x^\star)$, and

$$(A.5) \qquad \lambda_0 + \sum_{i \in I_0(x^\star)} \lambda_i = 1.$$

From (A.4), it exists that

$$\xi \in \lambda_0 \partial f(x^\star) + \sum_{i \in I_0(x^\star)} \lambda_i \partial g_i(x^\star),$$

such that

$$0 = \xi - \sum_{j \in I_l(x^\star)} \mu_j e_j + \sum_{h \in I_u(x^\star)} \mu_h e_h.$$

Then, recalling Definition 2.1 of $D(x)$, we have

$$(A.6) \qquad \xi^\top d \geq 0$$

for all $d \in D(x^\star)$. Thus, for all $d \in D(x^\star)$, we can write

$$\max \left\{ \xi^\top d : \xi \in \lambda_0 \partial f(x^\star) + \sum_{i \in I_0(x^\star)} \lambda_i \partial g_i(x^\star) \right\} \geq 0$$

with $\lambda_0 \geq 0$, $\lambda_i \geq 0$, $i \in I_0(x^\star)$, and, by (A.5), not all zero. This concludes the proof, choosing $\lambda_i = 0$, for all $i \notin I_0(x^\star)$. $\quad\square$

Taking into account the proof given above, in particular inequality (A.6), we can state the following result

LEMMA A.1. *Let $x^\star \in \mathcal{F}$ be a local minimum of the problem* (3.1). *Then, multipliers $\lambda_0^\star, \lambda_1^\star, \ldots, \lambda_m^\star \in \mathbb{R}$ not all zero, with*

$$\lambda_0^\star \geq 0, \quad \lambda_i^\star \geq 0 \quad \text{and} \quad \lambda_i^\star g_i(x^\star) = 0 \ \forall \ i = 1, \ldots, m,$$

and a vector $\bar{\xi} \in \lambda_0^\star \partial f(x^\star) + \sum_{i=1}^m \lambda_i^\star \partial g_i(x^\star)$ exist such that

$$\bar{\xi}^\top d \geq 0$$

for every $d \in D(x^\star)$.

Proof of Corollary 3.2. By assumption, in particular by condition (3.3), we know that $\bar{d} \in \mathbb{R}^n$ exists such that

$$(\xi^{g_i})^\top \bar{d} < 0, \forall\, \xi^{g_i} \in \partial g_i(x^\star) \quad \forall\, i \in I_0(x^\star),$$
$$-e_j^\top \bar{d} < 0, \forall\, j \in I_l(x^\star),$$
$$e_h^\top \bar{d} < 0, \forall\, h \in I_u(x^\star),$$

where $I_0(x^\star) = \{i : g_i(x^\star) = 0\}$, $I_l(x^\star) = \{j : x_j^\star = l_j\}$, and $I_u(x^\star) = \{h : x_h^\star = u_h\}$.

Now, by the alternative theorem in [42, Theorem 2.3.4] and [47], there cannot exist multipliers $\tilde{\lambda}_i \geq 0$, $i \in I_0(x^\star)$, $\tilde{\mu}_j \geq 0$, $j \in I_l(x^\star)$, $\tilde{\mu}_h \geq 0$, $h \in I_u(x^\star)$, with

$$(A.7) \qquad \sum_{i \in I_0(x^\star)} \tilde{\lambda}_i + \sum_{j \in I_l(x^\star)} \tilde{\mu}_j + \sum_{h \in I_u(x^\star)} \tilde{\mu}_h = 1,$$

such that

$$(A.8) \qquad 0 \in \sum_{i \in I_0(x^\star)} \tilde{\lambda}_i \partial g_i(x^\star) - \sum_{j \in I_l(x^\star)} \tilde{\mu}_j e_j + \sum_{h \in I_u(x^\star)} \tilde{\mu}_h e_h.$$

On the other hand, by Proposition 3.1, we know that multipliers $\lambda_0^\star \geq 0$, $\lambda_i^\star \geq 0$, with $\lambda_i^\star = 0$ when $g_i(x^\star) < 0$, exist such that (3.2) holds.

Then, we proceed by contradiction and assume that $\lambda_0^\star = 0$. Note that the multipliers $\lambda_i^\star$, $i = 1, \ldots, m$, cannot be all zero, since in this case all the multipliers would be zero, thus contradicting Proposition 3.1. Thus, we can define new multipliers

$$\bar{\lambda}_i = \lambda_i^\star / \beta, \qquad i \in I_0(x^\star),$$

where $\beta = \sum_{i \in I_0(x^\star)} \lambda_i^\star > 0$. Hence, we have that

$$(A.9) \qquad \bar{\lambda}_i \geq 0 \quad \text{and} \quad \sum_{i \in I_0(x^\star)} \bar{\lambda}_i = 1.$$

In this case, by Lemma A.1, there exists a vector

$$(A.10) \qquad \bar{\xi} \in \sum_{i=1}^m \bar{\lambda}_i \partial g_i(x^\star)$$

such that

$$(A.11) \qquad \bar{\xi}^\top d \geq 0$$

for every $d \in D(x^\star)$. Furthermore, by (A.11) the system

$$-\bar{\xi}^\top d > 0,$$
$$-e_j^\top d \leq 0 \qquad \forall\, j \in I_l(x^\star),$$
$$e_h^\top d \leq 0 \qquad \forall\, h \in I_u(x^\star)$$

does not have a solution, where the latter two sets of constraints imply $d \in D(x^\star)$. As a consequence, by the Farkas theorem (see, e.g., [36, Chapter 2]), we have that scalars not all zero $\alpha_j \geq 0$, $j \in I_l(x^\star)$, and $\alpha_h \geq 0$, $h \in I_u(x^\star)$, exist such that

$$(A.12) \qquad -\bar{\xi} = -\sum_{j \in I_l(x^\star)} \alpha_j e_j + \sum_{h \in I_u(x^\star)} \alpha_h e_h.$$

Now, by (A.9), we know that

$$\sigma = \sum_{i \in I_0(x^\star)} \bar\lambda_i + \sum_{j \in I_l(x^\star)} \alpha_j + \sum_{h \in I_u(x^\star)} \alpha_h \geq 1.$$

Then, we can define

$$\hat\lambda_i = \frac{\bar\lambda_i}{\sigma} \qquad \forall\, i \in I_0(x^\star),$$

$$\hat\alpha_j = \frac{\alpha_j}{\sigma} \qquad \forall\, j \in I_l(x^\star),$$

$$\hat\alpha_h = \frac{\alpha_h}{\sigma} \qquad \forall\, h \in I_u(x^\star),$$

thus having

(A.13)
$$\sum_{i \in I_0(x^\star)} \hat\lambda_i + \sum_{j \in I_l(x^\star)} \hat\alpha_j + \sum_{h \in I_u(x^\star)} \hat\alpha_h = 1.$$

Equations (A.10), (A.12), and (A.13) are in contradiction with the fact that there cannot exist multipliers such that (A.7) and (A.8) hold. □

Appendix B. Exactness properties of $Z_\varepsilon(x)$. In this section we first prove that any Clarke stationary point of problem (3.4) is stationary for problem (3.1). Then we give the proof of Proposition 3.6.

We begin by recalling, from [9], the definition of Clarke stationary point for a bound constrained problem, namely, a point $\bar x \in X$ such that

$$Z_\varepsilon^{Cl}(\bar x; d) \geq 0 \qquad \forall\, d \in D(\bar x).$$

Furthermore, we assume throughout this section that Assumption 1 holds.

PROPOSITION B.1. *Given problem* (3.1) *and considering problem* (3.4), *a threshold value $\varepsilon^\star > 0$ exists such that, for every $\varepsilon \in (0, \varepsilon^\star]$, the function $Z_\varepsilon(x)$ has no Clarke stationary points in $X \backslash \mathcal{F}$.*

Proof. We proceed by contradiction and assume that for any integer k an $\varepsilon_k \leq 1/k$ and a stationary point for problem (3.4) $x_k \in X \backslash \mathcal{F}$ exists. Then, let us consider a limit point $\bar x \in X \backslash \mathcal{F}$ of the sequence $\{x_k\}$ and let us relabel the corresponding subsequence $\{x_k\}$ again.

Since $\bar x \notin \mathcal{F}$, Assumption 1 guarantees that a direction $\bar d \in D(\bar x)$ exists such that

$$(\xi^{g_i})^\top \bar d < 0 \qquad \forall\, \xi^{g_i} \in \partial g_i(\bar x),\ i \in \{1, \dots, m : g_i(\bar x) \geq 0\}.$$

In particular, it holds that

$$(\xi^{g_i})^\top \bar d < 0 \qquad \forall\, \xi^{g_i} \in \partial g_i(\bar x),\ i \in I(\bar x),$$

where $I(\bar x) = \{i \in \{1, \dots, m\} : g_i(\bar x) = \phi(\bar x)\}$ and $\phi(x) = \max\{0, g_1(x), \dots, g_m(x)\}$, with $\phi(\bar x) > 0$ since $\bar x \notin \mathcal{F}$. The above property can be equivalently expressed by saying that a positive scalar η exists such that

(B.1)
$$\max_{\substack{\xi^{g_i} \in \partial g_i(\bar x) \\ i \in I(\bar x)}} (\xi^{g_i})^\top \bar d = -\eta < 0.$$

Recalling that for k sufficiently large, $D(\bar{x}) \subseteq D(x_k)$ (see, e.g., [29]), so that $\bar{d} \in D(x_k)$, we get, by considering that x_k is a Clarke stationary point of problem (3.4), that

$$(B.2) \qquad Z_\varepsilon^{Cl}(x_k; \bar{d}) \geq 0.$$

By [9, Proposition 2.1.2], $Z_\varepsilon^{Cl}(x; \bar{d}) = \max_{\xi \in \partial Z_\varepsilon(x)} \xi^\top \bar{d}$, and we know that

$$\partial Z_\varepsilon(x) \subseteq \partial f(x) + \frac{1}{\varepsilon} \partial(\max\{0, g_1(x), \ldots, g_m(x)\})$$

and (see [9, Proposition 2.3.12])

$$\partial(\max\{0, g_1(x), \ldots, g_m(x)\}) \subseteq Co\left(\{\partial g_i(x) : i \in I(x)\}\right) = \bigcup\left\{\sum_{i \in I(x)} \beta^i \partial g_i(x)\right\},$$

where $\beta^i \geq 0$, $i \in I(x)$, and $\sum_{i \in I(x)} \beta^i = 1$. Hence, β_k^i, $i \in I(x_k)$, exist such that (B.2) can be written as

$$(B.3) \qquad \left(\xi_k^f + \frac{1}{\varepsilon_k} \sum_{i \in I(x_k)} \beta_k^i \xi_k^{g_i}\right)^\top \bar{d} \geq 0,$$

$$\sum_{i \in I(x_k)} \beta_k^i = 1, \qquad \beta_k^i \geq 0,$$

for some $\xi_k^f \in \partial f(x_k)$, $\xi_k^{g_i} \in \partial g_i(x_k)$.

Now, recalling that m is a finite number, we can consider the subsequence of $\{x_k\}$ where $I(x_k) = \bar{I}$.

Then, since the generalized gradient of a locally Lipschitz continuous function is locally bounded, it results that all the considered sequences $\{\xi_k^f\}$, $\{\xi_k^{g_i}\}$, $i \in \bar{I}$, where $\xi_k^f \in \partial f(x_k)$, $\xi_k^{g_i} \in \partial g_i(x_k)$, $x_k \in X$, are bounded.[1] Hence, we get that

$$(B.4a) \qquad \xi_k^f \to \bar{\xi}^f,$$
$$(B.4b) \qquad \xi_k^{g_i} \to \bar{\xi}^{g_i} \qquad \forall\, i \in \bar{I},$$
$$(B.4c) \qquad \beta_k^i \to \bar{\beta}^i \qquad \forall\, i \in \bar{I}.$$

Further, since ∂f and ∂g_i, $i \in \bar{I}$, are upper semicontinuous at $\bar{x}$ (see Proposition 2.1.5 in [9]), it results that $\bar{\xi}^f \in \partial f(\bar{x})$, $\bar{\xi}^{g_i} \in \partial g_i(\bar{x})$, $i \in \bar{I}$.

Now, since by continuity of the problem functions we have for k sufficiently large

$$\{i : g_i(\bar{x}) - \phi(\bar{x}) < 0\} \subseteq \{i : g_i(x_k) - \phi(x_k) < 0\},$$

it results, for k sufficiently large,

$$\{i : g_i(x_k) - \phi(x_k) = 0\} = I(x_k) \subseteq I(\bar{x}) = \{i : g_i(\bar{x}) - \phi(\bar{x}) = 0\},$$

so that

$$(B.5) \qquad \bar{I} \subseteq I(\bar{x}).$$

---

[1] This result follows by considering that a finite covering of X by bounded sets exists and that any ξ_k^f, $\xi_k^{g_i}$, $i \in \bar{I}$, are bounded on the latter sets.

Then, by (B.1), (B.4), and (B.5), we get, for k sufficiently large,

$$(\text{B.6}) \qquad\qquad \left(\xi_k^{g_i}\right)^\top \bar{d} \leq -\frac{\eta}{2} \qquad \forall\, i \in \bar{I}.$$

Now, by multiplying (B.3) by ε_k we have

$$\left(\varepsilon_k \xi_k^f + \sum_{i \in \bar{I}} \beta_k^i \xi_k^{g_i}\right)^\top \bar{d} \geq 0,$$

which, by (B.6), yields

$$0 \leq \left(\varepsilon_k \xi_k^f + \sum_{i \in \bar{I}} \beta_k^i \xi_k^{g_i}\right)^\top \bar{d} \leq \left(\varepsilon_k \xi_k^f\right)^\top \bar{d} - \frac{\eta}{2}.$$

Finally, the above relation, considering (B.4a), gives raise to a contradiction when $\varepsilon_k \to 0$. $\quad\square$

Now we report three further results concerning the exactness of $Z_\varepsilon(x)$ from [15].

PROPOSITION B.2. *A threshold value $\varepsilon^\star > 0$ exists such that for any $\varepsilon \in (0, \varepsilon^\star]$, every local minimum point of problem* (3.4) *is also a local minimum point of problem* (3.1).

PROPOSITION B.3. *A threshold value $\varepsilon^\star > 0$ exists such that for any $\varepsilon \in (0, \varepsilon^\star]$, every global minimum point of problem* (3.4) *is also a global minimum point of problem* (3.1), *and conversely.*

In order to give stationarity results for problem (3.4), we have the following proposition.

PROPOSITION B.4. *For any $\varepsilon > 0$, every Clarke stationary point $\bar{x}$ of problem* (3.4), *such that $\bar{x} \in \mathcal{F}$, is also a stationary point of problem* (3.1).

Proof. Since $\bar{x}$ is, by assumption, a Clarke stationary point of problem (3.4), then, by definition of Clarke stationarity, we know that for all $d \in D(\bar{x})$,

$$\max\left\{\xi^\top d : \xi \in \partial Z_\varepsilon(\bar{x})\right\} \geq 0,$$

that is, for all $d \in D(\bar{x})$ there exists $\xi_d \in \partial Z_\varepsilon(\bar{x})$ such that $(\xi_d)^\top d \geq 0$. Now, we recall that

$$\partial Z_\varepsilon(x) \subseteq \partial f(x) + \frac{1}{\varepsilon} \sum_{i \in I(x)} \beta_i \partial g_i(x)$$

for some β_i, $i \in I(x)$, such that $\sum_{i \in I(x)} \beta_i = 1$ and $\beta_i \geq 0$ for all $i \in I(x)$. Hence, we have that $\xi_d \in \partial f(\bar{x}) + \frac{1}{\varepsilon} \sum_{i \in I(\bar{x})} \beta_i \partial g_i(\bar{x})$. Then, denoting $\lambda_i = \beta_i/\varepsilon$, $i \in I(\bar{x})$, we can write for all $d \in D(\bar{x})$,

$$\max\left\{\xi^\top d : \xi \in \partial f(\bar{x}) + \sum_{i \in I(\bar{x})} \lambda_i \partial g_i(\bar{x})\right\} \geq 0,$$

with $\lambda_i \geq 0$. The above condition, along with $\bar{x} \in \mathcal{F}$, proves stationarity of $\bar{x}$ for problem (3.1) and concludes the proof. $\quad\square$

Finally, we can prove Proposition 3.6.

Proof of Proposition 3.6. Since $\bar{x}$ is Clarke–Jahn stationary for problem (3.4), we have, by Definition 2.3,

(B.7)
$$Z_\varepsilon^\circ(\bar{x}; d) \geq 0 \quad \forall\, d \in D(\bar{x}).$$

Then, by (1.1) and (2.2), we have that

$$\limsup_{y \to \bar{x},\, t \downarrow 0} \frac{Z_\varepsilon(y + td) - Z_\varepsilon(y)}{t} = Z_\varepsilon^{Cl}(\bar{x}; d) \geq Z_\varepsilon^\circ(\bar{x}; d) \quad \forall\, d \in D(\bar{x}),$$

which, by (B.7), gives

$$Z_\varepsilon^{Cl}(\bar{x}; d) \geq 0 \quad \forall\, d \in D(\bar{x}).$$

Now, the proof follows by considering Propositions B.1 and B.4.   □

Acknowledgments. We are indebted to three anonymous reviewers and the associate editor, whose suggestions and stimulating comments greatly helped us improve the quality of the paper. In particular, the comments of one reviewer and a counterexample of another reviewer spotted a theoretical bug of the paper and forced us to investigate more deeply some basic aspects of our proposal.

REFERENCES

[1] M. A. ABRAMSON, C. AUDET, G. COUTURE, J. E. DENNIS, JR., S. LE DIGABEL, AND C. TRIBES, *The NOMAD Project*, http://www.gerad.ca/nomad.

[2] M. A. ABRAMSON, C. AUDET, J. E. DENNIS, JR., AND S. LE DIGABEL, *Orthomads: A deterministic MADS instance with orthogonal directions*, SIAM J. Optim., 20 (2009), pp. 948–966.

[3] N. ALEXANDROV AND M. Y. HUSSAINI, *Multidisciplinary Design Optimization: State of the Art*, SIAM, Philadelphia, 1997.

[4] C. AUDET, A. CUSTÓDIO, AND J. E. DENNIS, JR., *Erratum: Mesh adaptive direct search algorithms for constrained optimization*, SIAM J. Optim., 18 (2008), pp. 1501–1503.

[5] C. AUDET AND J. E. DENNIS, JR., *Mesh adaptive direct search algorithms for constrained optimization*, SIAM J. Optim., 17 (2006), pp. 188–217.

[6] C. AUDET AND J. E. DENNIS, JR., *A progressive barrier for derivative-free nonlinear programming*, SIAM J. Optim., 20 (2009), pp. 445–472.

[7] C. AUDET, J. E. DENNIS, JR., AND S. LE DIGABEL, *Globalization strategies for mesh adaptive direct search*, Comput. Optim. Appl., 46 (2010), pp. 193–215.

[8] P. BRATLEY AND B. FOX, *Algorithm* 659: *Implementing Sobol's quasirandom sequence generator*, ACM Trans. Math. Softw., 14 (1988), pp. 88–100.

[9] F. H. CLARKE, *Optimization and Nonsmooth Analysis*, John Wiley & Sons, New York, 1983.

[10] A. R. CONN, K. SCHEINBERG, AND L. N. VICENTE, *Global convergence of general derivative-free trust-region algorithms to first and second order critical points*, SIAM J. Optim., 20 (2009), pp. 387–415.

[11] A. CONN, K. SCHEINBERG, AND L. N. VICENTE, *Introduction to Derivative-Free Optimization*, MPS/SIAM Ser. Optim., SIAM, Philadelphia, 2009.

[12] A. L. CUSTÓDIO, J. F. A. MADEIRA, A. I. F. VAZ, AND L. N. VICENTE, *Direct multisearch for multiobjective optimization*, SIAM J. Optim., 21 (2011), pp. 1109–1140.

[13] A. L. CUSTÓDIO AND L. N. VICENTE, *Using sampling and simplex derivatives in pattern search methods*, SIAM J. Optim., 18 (2007), pp. 537–555.

[14] A. L. CUSTÓDIO, J. E. DENNIS, JR., AND L. N. VICENTE, *Using simplex gradients of nonsmooth functions in direct search methods*, IMA J. Numer. Anal., 28 (2008), pp. 770–784.

[15] G. DI PILLO AND F. FACCHINEI, *Exact barrier function methods for Lipschitz programs*, Appl. Math. Optim., 32 (1995), pp. 1–31.

[16] E. D. DOLAN, R. M. LEWIS, AND V. TORCZON, *On the local convergence of pattern search*, SIAM J. Optim., 14 (2003), pp. 567–583.

[17] R. FLETCHER AND S. LEYFFER, *Nonlinear programming without a penalty function*, Math. Program., 91 (2002), pp. 239–269.

[18] L. GRIPPO, F. LAMPARIELLO, AND S. LUCIDI, *Global convergence and stabilization of uncon- strained minimization methods without derivatives*, J. Optim. Theory Appl., 56 (1988), pp. 385–406.

[19] J. H. HALTON, *On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals*, Numer. Math., 2 (1960), pp. 84–90.

[20] W. L. HARE, *Using derivative free optimization for constrained parameter selection in a home and community care forecasting model*, in International Perspectives on Operations Re- search and Health Care, J. Blake and M. Carter, eds., Proceedings of the 34th Meeting of the EURO Working Group on Operational Research Applied to Health Sciences, 2010, pp. 61–73.

[21] J. B. HIRIART-URRUTY, *On optimality conditions in nondifferentiable programming*, Math. Program., 14 (1978), pp. 73–86.

[22] J. JAHN, *Introduction to the Theory of Nonlinear Optimization*, Springer-Verlag, Berlin, 1996.

[23] N. KARMITSA, *Test Problems for Large-Scale Nonsmooth Minimization*, Tech. report B. 4/2007, Department of Mathematical Information Technology, University of Jyväskylä, Finland, 2007.

[24] T. G. KOLDA, R. M. LEWIS, AND V. TORCZON, *Optimization by direct search: New perspectives on some classical and modern methods*, SIAM Rev., 45 (2003), pp. 385–482.

[25] T. G. KOLDA, R. M. LEWIS, AND V. TORCZON, *A Generating Set Direct Search Augmented Lagrangian Algorithm for Optimization with a Combination of General and Linear Con- straints*, Tech. report SAND2006-5315, Sandia National Laboratories, 2006.

[26] T. G. KOLDA, R. M. LEWIS, AND V. TORCZON, *Stationarity results for generating set search for linearly constrained optimization*, SIAM J. Optim., 17 (2007), pp. 943–968.

[27] S. LE DIGABEL, *Algorithm 909: NOMAD: Nonlinear optimization with the MADS algorithm*, ACM Trans. Math. Softw., 37 (2011), pp. 1–15.

[28] R. M. LEWIS AND V. TORCZON, *A globally convergent augmented Lagrangian pattern search algorithm for optimization with general constraints and simple bounds*, SIAM J. Optim., 12 (2002), pp. 1075–1089.

[29] C. J. LIN, S. LUCIDI, L. PALAGI, A. RISI, AND M. SCIANDRONE, *Decomposition algorithm model for singly linearly-constrained problems subject to lower and upper bounds*, J. Optim. Theory Appl., 141 (2009), pp. 107–126.

[30] G. LIUZZI AND S. LUCIDI, *New results on a continuously differentiable exact penalty function*, SIAM J. Optim., 2 (1992), pp. 558–574.

[31] G. LIUZZI, S. LUCIDI, AND F. RINALDI, *Derivative-free methods for bound constrained mixed- integer optimization*, Comput. Optim. Appl., 53 (2012), pp. 505–526.

[32] G. LIUZZI, S. LUCIDI, AND M. SCIANDRONE, *A derivative-free algorithm for linearly constrained finite minimax problems*, SIAM J. Optim., 16 (2006), pp. 1054–1075.

[33] G. LIUZZI, S. LUCIDI, AND M. SCIANDRONE, *Sequential penalty derivative-free methods for nonlinear constrained optimization*, SIAM J. Optim., 20 (2010), pp. 2614–2635.

[34] S. LUCIDI AND M. SCIANDRONE, *A derivative-free algorithm for bound constrained optimization*, Comput. Optim. Appl., 21 (2002), pp. 119–142.

[35] S. LUCIDI AND M. SCIANDRONE, *On the global convergence of derivative free methods for un- constrained optimization*, SIAM J. Optim., 13 (2002), pp. 97–116.

[36] O. L. MANGASARIAN, *Nonlinear Programming*, Classics in Applied Mathematics, SIAM, Philadelphia, 1994.

[37] J. J. MORÉ AND S. M. WILD, *Benchmarking derivative-free optimization algorithms*, SIAM J. Optim., 20 (2009), pp. 172–191.

[38] G. DI PILLO AND S. LUCIDI, *An augmented Lagrangian function with improved exactness prop- erties*, SIAM J. Optim., 12 (2001), pp. 376–406.

[39] M. J. D. POWELL, *A direct search optimization method that models the objective and constraint functions by linear interpolation*, in Advances in Optimization and Numerical Analysis, S. Gomez and J.-P. Hennart, eds., Kluwer Academic Publishers, Dordrecht, Netherlands, 1994, pp. 51–67.

[40] M. J. D. POWELL, *The BOBYGA Algorithm for Bound Constrained Optimization Without Derivatives*, Tech. report, Department of Applied Mathematics and Theoretical Physics, University of Cambridge, UK, 2009.

[41] R. T. ROCKAFELLAR, *Convex Analysis*, Princeton Mathematical Series, Princeton University Press, Princeton, NJ, 1970.

[42] K. SHIMIZU, Y. YSHIZUKA, AND J. F. BARD, *Nondifferentiable and Two-Level Mathematical Programming*, Kluwer Academic Publishers, Norwell, MA, 1997.

[43] I. SOBOL, *Uniformly distributed sequences with an additional uniform property*, USSR Comput. Math. Math. Phys., 16 (1977), pp. 236–242.

[44] V. Torczon, *On the convergence of pattern search algorithms*, SIAM J. Optim., 7 (1997), pp. 1–25.

[45] J. Vlček and V. Lukšan, *Test Problems for Nonsmooth Unconstrained and Linearly Constrained Optimization*, Tech. report, Institute of Computer Science, Academy of Sciences of the Czech Republic, 2000.

[46] L. N. Vicente and A. L. Custódio, *Analysis of direct searches for discontinuous functions*, Math. Program., 133 (2012), pp. 299–325.

[47] Y. Yshizuka and K. Shimizu, *Necessary and sufficient conditions for the efficient solutions of nondifferentiable multi-objective problems*, IEEE Trans. Systems Man Cybernet., 14 (1984), pp. 624–629.